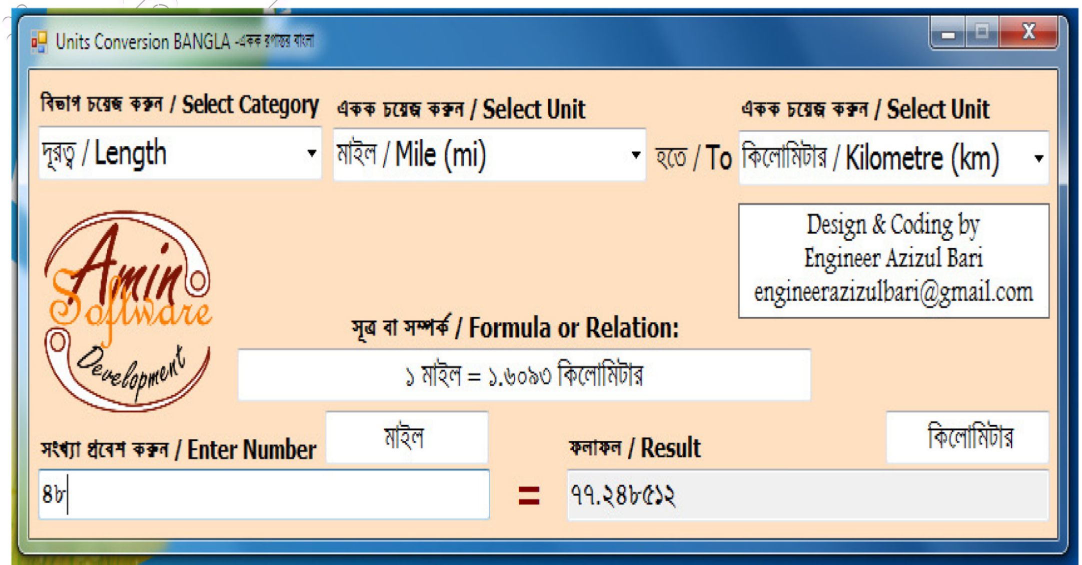
VB.NET

ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং

নতুন শিক্ষার্থীদের জন্য গুরুত্বপূর্ণ প্রশিক্ষণ রিসোর্স

দ্বিতীয় খণ্ড

Part Two



বিশেষ ইন্টারনেট সংস্করণ

Special Internet Edition

ইঞ্জিনিয়ার আজিজুল বারী
engineerazizulbari@gmail.com

খানক্বায়ে আমীনিয়া-আসগরিয়া গবেষণা বিভাগ
Ali Centre, Subidbazar, Sylhet, BD

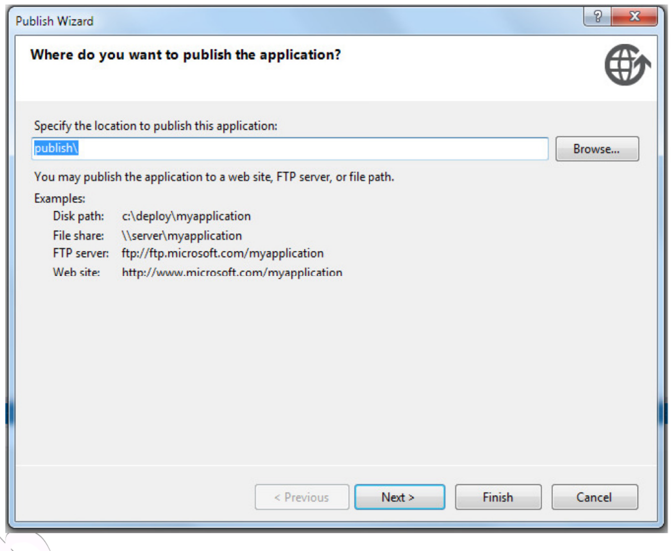
১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

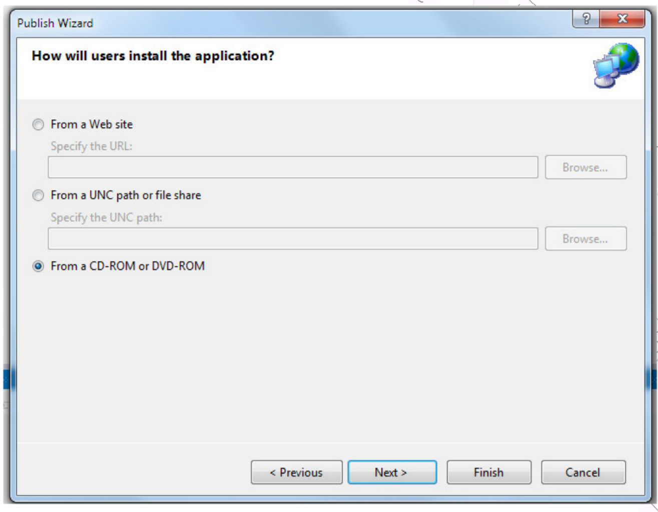
বিস্তারণ বা প্রকাশ করতে নিচের পদক্ষেপগুলো ধারাবাহিকভাবে অনুসরণ করুন:

১. বিল্ড [Build] মেনুতে যেয়ে চয়েজ করুন 'পাবলিশ ইউনিট্স কনভার্শন বাংলা'। নিচের উইজার্ড জানালাটি আসবে।

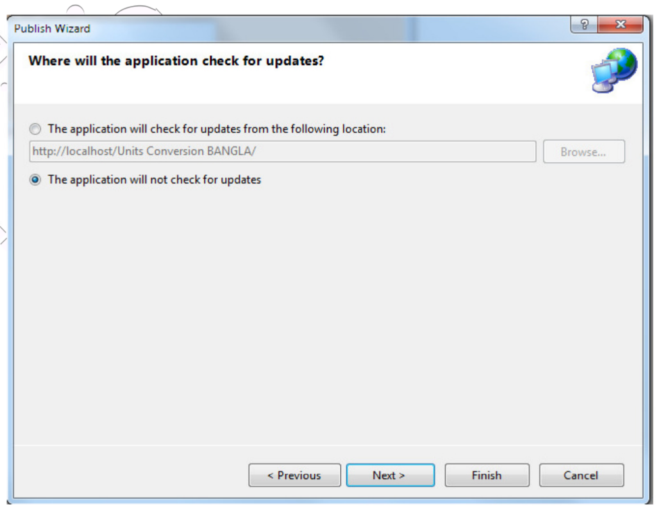
২. ব্রাউজ বোতামে ক্লিক করে কোথায় এটি পাবলিশ করবেন জানিয়ে দিন। নিচে কিছু উদাহরণ দেওয়া আছে। সুতরাং আমার উইজার্ডে দেখাবে এই প্যাথটি:
C:\Users\htci\Desktop\Bangla Units Conversion



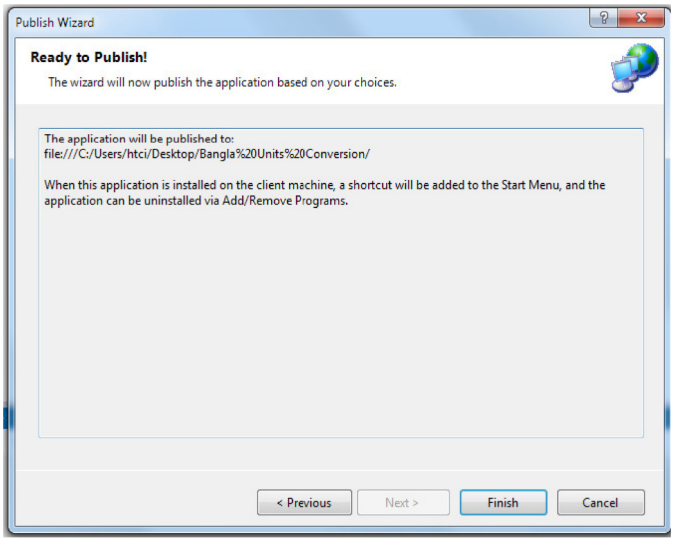
৩. এবার নেক্সট বোতাকে ক্লিক করুন। উইজার্ড এবার নিচের জানালাটি প্রতর্শন করবে। সর্বশেষ রেডিও বোতাম চয়েজ করা থাকবে। সম্ভবত আপনি এ চয়েজই করবেন।



এরপর নেক্সট বোতামে টিপ দিন। নিচের জানালা আসবে। এ্যাপলিকেশন আপডেইট কোথায় করবে? এ প্রশ্নের জবাবে আপনি চয়েজ করবেন নিচের রেডিও বোতাম। অর্থাৎ



আপডেইট করবে না। এবার পুনরায় নেক্সট বোতামে ক্লিক করুন। নিচের ডায়লগটি আসবে। আপনার এ্যাপলিকেশন



পাবলিশের জন্য প্রস্তুত হয়ে গেছে! জানালায় কিছু তথ্য থাকবে: The application will be published to: file:///C:/Users/htci/Desktop/Bangla%20Units%20Conversion/
When this application is installed on the client machine, a shortcut will be added to the Start Menu, and the application can be uninstalled via Add/Remove Programs.। দেখতেই পাচ্ছেন- এতে আছে ফাইল প্যাথ। আরো আছে ইনস্টল করার পর স্টার্ট মেনুতে যে শর্টকাট আইকন থাকবে সে তথ্য।

৪. এবার ফিনিশ [Finish] বোতামে টিপ দিন। আপনার এ্যাপলিকেশন পাবলিশ হয়ে যাবে। ব্যস্। বন্ধুদেরকে এটি দেখিয়ে বাহাদুরী দেখাবার পালা এখন!

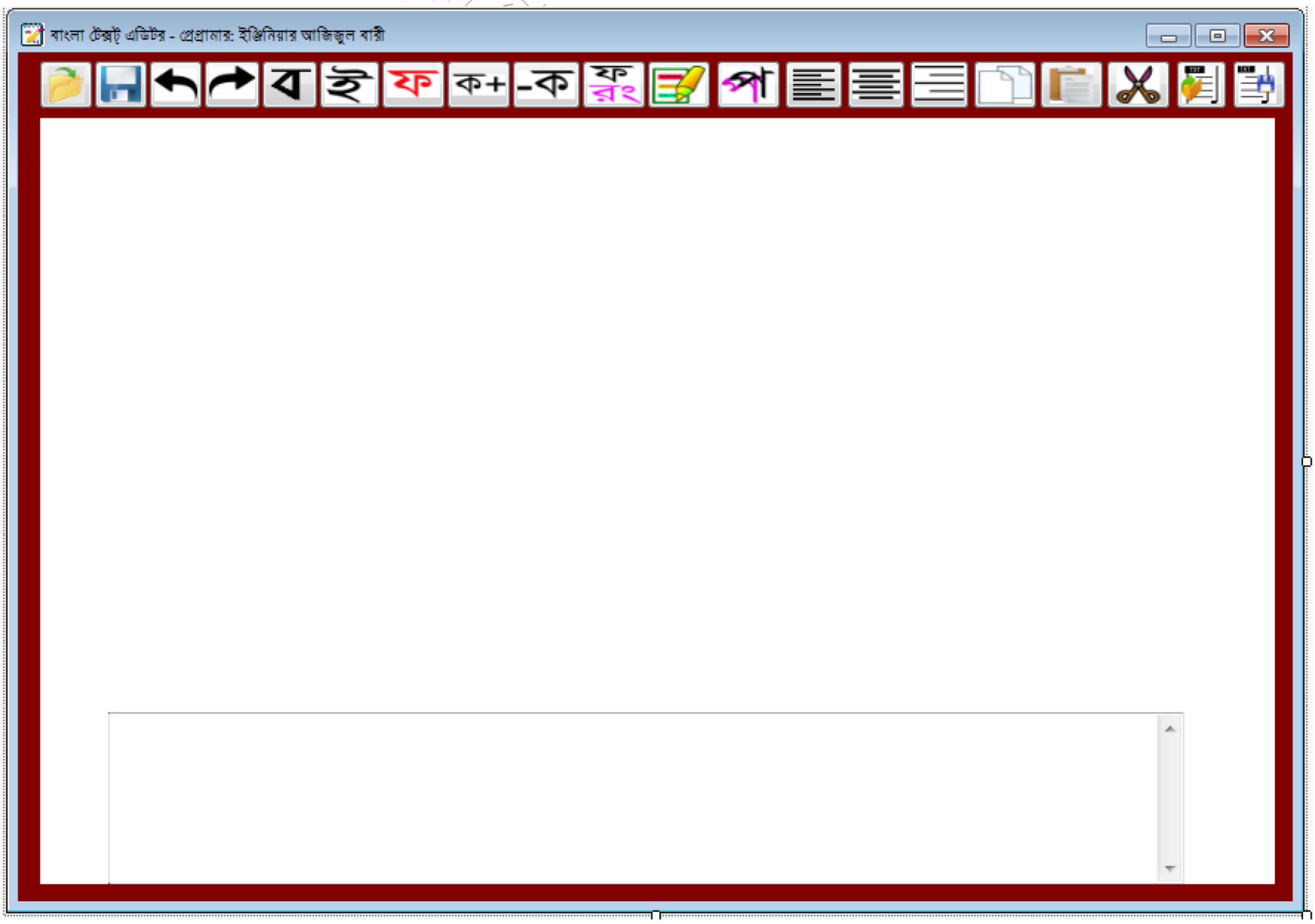
১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

Application 4 : Bangla / English Word Processor - বাংলা / ইংরেজি ওয়ার্ড প্রসেসর।

হ্যাঁ, ভিজুয়্যাল বেসিক.কম দ্বারা ওয়ার্ড প্রসেসর তৈরি করা যায়- আর তা আসলে খুব কঠিন কিছু নয়! অসংখ্য ওয়ার্ড প্রসেসর থাকাসত্ত্বেও আরেকটি কেনো? জবাবে বলা যায়, কেনো নয়? আমরা তো শিখবো- ঠিক না? তাছাড়া বাংলার ওপর গুরুত্ব দিয়ে এটি আমরা তৈরি করতে চাই। সুতরাং আসুন, গড়ে তুলি আমাদের চতুর্থ এ প্রোগ্রামটি- বাংলা-ইংরেজি ওয়ার্ড প্রসেসর।

১. ইউআই জিডাইন: প্রথমেই আমরা ডিজাইন করবো একটি আকর্ষণীয় ইউজার ইন্টারফেইস। নিচের চিত্রে একটি স্যাম্পুল দেখিয়েছি। শিক্ষার্থীরা চাইলে অন্যভাবেও ডিজাইন করতে পারেন। বরং চেষ্টা করবেন, নিজের মনের মতো ডিজাইন করতে। তবে মনে রাখবেন, নিম্লে তালিকাভুক্ত বস্তুগুলো ফর্মে থাকতে হবে।



(ক) একটি উইন্ডোজ ফর্ম:

(১) টেক্সট্ : "বাংলা টেক্সট্ এডিটর" [আমি নিজের নামটিও লিখেছি। আপনি আপনার নাম লিখতে পারেন]। (২) অটোস্কেলমোড [AutoScaleMode] = "Font"। (৩) পেছনের রং = Maroon। (৪) ফন্ট = "Times New Roman, 14pt]"। (৫) সম্ভব হলো একটি 'প্রোগ্রাম' আইকোন। [আমি একটি ও বাকী সব আইকোন নিজে ডিজাইন করে দিয়েছি। সব আইকোন ছবি প্রজেক্টে এ্যামবেড [যুক্ত] করেছি। কিভাবে করা যায়- তা একটু পরই লিখে দেবো। (৬) ফর্মের সাইজ = ৮৭৬, ৬০৮। ব্যস! বাকি সব প্রোপ্রাটি যেভাবে আছে সেভাবেই থাকবে।

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

(খ) ২০টি ছোট্ট সাইজের বোতাম [বাম থেকে]: (১) OpenFileBt, (২) SaveFileBt, (৩) UndoBt, (৪) RedoBt, (৫) BoldBt, (৬) ItalicBt, (৭) FontBT, (৮) SizeUpBt, (৯) SizeDnBt, (১০) FontColBt, (১১) HighLiteColBt (১২) BckColBt, (১৩) AlinLBt, (১৪) AlinCBt, (১৫) AlinRBt, (১৬) CopyBt, (১৭) PstBt, (১৮) CutBt, (১৯) OpenTxtBt, (২০) SaveTxtBt। এগুলো সবই বোতামের নাম (name) প্রোপার্টি হবে। কোডে আমরা এসব নাম দ্বারাই বোতামগুলোকে সম্বোধন করবো। আপনি চাইলে আইকোন যুক্ত করতে পারেন। তবে সংক্ষেপে বাংলা বা ইংরেজি Text প্রেপার্টিও লিখতে পারেন। আইকোন দিলে বোতামের BackColour হবে ট্রান্সপ্যারেন্ট। এর পরের প্রেপার্টি তথা BackgroundImage প্রোপার্টিতে আইকোনের ছবি দেবেন। আর প্রতিটি বোতামের ফন্ট হবে Times New Roman, 14pt।

(গ) একটি RichTextBox1 কন্ট্রোল। টুলবাক্স চয়েজ করে অলউইন্ডোজ ফর্মস্ ট্যাবে ক্লিক করলেই নিচে দেখতে পাবেন এই কন্ট্রোলটি। এই কন্ট্রোলের বৈশিষ্ট্য হলো: একই সময় বিভিন্ন ফন্ট, সাইজ, রঙ ইত্যাদি ফরমাটে লেখা যায়। সুতরাং আমরা এতে বাংলা ও ইংরেজি লিখতে পারবো অনায়াসেই।

(ঙ) একটি লুকানো টেক্স বাক্স: (১) এর ফন্ট প্রোপার্টি হবে TonnyBanglaMJ, 14pt [বা বিজয়ের অন্য কোনো ফন্ট হলেও চলবে]। (২) এর মাল্টিলাইন প্রোপার্টি হবে True। (৩) স্ক্রলবার্স প্রোপার্টি হবে Vertical।

ব্যস! আর কিছু নেই। একবার প্রোগ্রাম রান করে দেখে নিন। আপনি চাইলে টেক্সট বাক্সে বাংলা বা ইংরেজিতে লিখতেও পারবেন। তবে আমরা তো চাই আরো কিছু ফাংশন্যালিটি! তাই আসুন প্রতিটি বোতামের কোড কি হবে দেখে নিই। প্রথমে পুরো কোডই একসাথে তুলে ধরছি। এরপর বিভিন্ন সাবরুটিনের ব্যাখ্যা একে একে তুলে ধরবো।

প্রোগ্রামের কোড:

```
Public Class Form1
  'কোডিং: ইঞ্জিনিয়ার আজিজুল বারী: engineerazizulbari@gmail.com
  Dim RichTextBoxSize As Integer = 838 'এই ইন্টেজার গ্লেবাল দ্বারা টেক্সট বাক্সের সাইজ নিয়ন্ত্রণ করা হবে।
  'ফর্ম লোড ইভেন্ট সাবের কোড।
  Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    'টেক্সট বাক্সের সাইজ সেট করা।
      RichTextBox1.Size = New Point(RichTextBoxSize, RichTextBox1.Height)
    'ফর্মের মাঝখানে টেক্সট বাক্স নিয়ে আসা।
      RichTextBox1.Location = New Point((Me.Size.Width - RichTextBoxSize) / 2, RichTextBox1.Location.Y)
      'কন্টেক্সট মেনু দেখানোর জন্য নিচে ডিফাইন করা সাবটি কল করা। টেক্সট বাক্সের ভেতর রাইট-ক্লিক করলে এই মেনু আসবে।
       InitializeMyContextMenu() 'নিচের সাবরুটিন কল।
  End Sub
    'এই সাবই কনটেক্সট মেনু প্রদর্শন করবে যখনই ব্যবহারকারী রাইট-ক্লিক করবেন টেক্সট বাক্সের ভেতর।
  Private Sub InitializeMyContextMenu()
          'কনটেক্সট মেনু ও মেনু আইটেম সৃষ্টি করা।
    Dim contextMenu1 As New ContextMenu() 'প্রথমে একটি নতুন কনটেক্সট মেনু ডিফাইন করতে হবে।
    Dim menuItem1 As New MenuItem("&কাটুন") 'এবার আরেকটি নতুন মেনু আইটেম ডিফাইন করে তাতে লেখা দিতে হবে।
    AddHandler menuItem1.Click, AddressOf CutBt_Click 'কোন বোমাত তা হ্যান্ডল করবে, জানাতে হবে।
    Dim menuItem2 As New MenuItem("&কপি")
       AddHandler menuItem2.Click, AddressOf CopyBt_Click
         Dim menuItem3 As New MenuItem("&পেস্ট")
     AddHandler menuItem3.Click, AddressOf PstBt_Click
      'মেনু আইটেমস্ প্রোপার্টি ব্যবহার করে এড মেথড ডাকতে হবে।
    'ফলে মেনু আইটেমগুলো মূল মেনু আইটেম কালেকশনে সংযুক্ত হবে।
    contextMenu1.MenuItems.Add(menuItem1)
    contextMenu1.MenuItems.Add(menuItem2)
    contextMenu1.MenuItems.Add(menuItem3)
    'mainMenu1-কে রিচটেক্সট্বাক্সের সঙ্গে এসাইন করতে হবে।
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        RichTextBox1.ContextMenu = contextMenu1
    End Sub
        'ফর্মের রিসাইজ ইভেন্ট সাবের কোড। ফর্মের সাইজ পরিবর্তন হলে রিচ টেক্সট বাক্সের সাইজ ও পরিবর্তন হতে হবে।
    Private Sub Create_Resize(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Resize
        If Me.Size.Width > RichTextBoxSize + 50 Then 'যদি ফর্মের প্রস্থ সাইজ হয় বাক্সের সাইজ + ৫০ পয়েন্ট থেকে বেশি-
            RichTextBox1.Width = Me.Size.Width - 100 'তাহলে বাক্সের প্রস্থ সাইজ হবে একশ পয়েন্ট কম।
            RichTextBox1.Anchor = AnchorStyles.None 'কন্টেইনারের ভেতর কোনো বিশেষ স্টাইলে টেক্সট বাক্স থাকবে না।
            RichTextBox1.Anchor = AnchorStyles.Top Or AnchorStyles.Bottom 'শুধু উপর ও নিচের পার্শ্বের নিকটে থাকবে।
            'নিচের লাইনটির কাজ হলো ফর্মের সেন্টারে টেক্স বাক্স থাকার নিশ্চয়তা দেওয়া- সর্বদা।
            RichTextBox1.Location = New Point((Me.Size.Width - (Me.Size.Width - 100)) / 2, RichTextBox1.Location.Y)
        End If
        'নিচের কোড দ্বারা ফর্মের ভেতর টেক্সড্ বাক্সের অবস্থান, প্রস্থ ও আনকর নিয়ন্ত্রণ করা হয়েছে।
            If Me.Size.Width <= RichTextBoxSize + 50 Then
          If RichTextBox1.Location.X < 10 Then
            RichTextBox1.Location = New Point(5, RichTextBox1.Location.Y)
          End If
          If RichTextBox1.Width + 25 > Me.Size.Width Then
            RichTextBox1.Size = New Point(Me.Size.Width - 10 - RichTextBox1.Location.X, RichTextBox1.Size.Height)
          End If
          RichTextBox1.Anchor = AnchorStyles.None
          RichTextBox1.Anchor = AnchorStyles.Top Or AnchorStyles.Bottom Or AnchorStyles.Left Or AnchorStyles.Right
        End If
    End Sub
    'এটা আনডু বোতামের কোড।
    Private Sub UndoBt_Click(sender As Object, e As EventArgs) Handles UndoBt.Click
        RichTextBox1.Undo() 'যা এই মাত্র করা হয়েছে তা আনডু করবে।
        RichTextBox1.Focus()
    End Sub
    'এটা রিডু বোতামের কোড।
    Private Sub RedoBt_Click(sender As Object, e As EventArgs) Handles RedoBt.Click
        RichTextBox1.Redo() 'যা এই মাত্র আনডু হয়েছে তা রিডু করবে।
        RichTextBox1.Focus()
    End Sub
    'এটা বল্ড তৈরির কোড।
    Private Sub BoldBt_Click(sender As Object, e As EventArgs) Handles BoldBt.Click
        If RichTextBox1.SelectionFont.Bold = True Then 'যদি বল্ড থেকে তাকে তাহলে নরম্যাল বানাতে হবে।
            If RichTextBox1.SelectionFont.Italic = True Then 'ইটালিক থাকলে সেটা সহ নরম্যাল করতে হবে- যা নিচের কোড লাইনের কাজ।
                RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Regular +
FontStyle.Italic)
            Else 'অন্যথায় ইটালিক ছাড়াই একই কাজ করতে হবে।
                RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Regular)
            End If
        ElseIf RichTextBox1.SelectionFont.Bold = False Then 'যদি বল্ড না থাকে তাহলে বল্ড করতে হবে।
            If RichTextBox1.SelectionFont.Italic = True Then 'ইটালিক থাকলে সেটা সহ বল্ড করতে হবে- যা নিচের কোড লাইনের কাজ।
                RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Bold + FontStyle.Italic)
            Else 'অন্যথায় ইটালিক ছাড়াই একই কাজ করতে হবে।
                RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Bold)
            End If
        End If
        RichTextBox1.Focus() 'ফেকাস হবে টেক্সট বাক্সে।
    End Sub
    'এটা হলো ইটালিক বোতামের কোড।
Private Sub ItalicBt_Click(sender As Object, e As EventArgs) Handles ItalicBt.Click
        If RichTextBox1.SelectionFont.Italic = True Then 'ইটালিক থাকলে নরম্যাল হবে।
            If RichTextBox1.SelectionFont.Bold = True Then 'বল্ড আছে কি? থাকলে নিচের লাইন নিশ্চিত করবে বল্ডসহ নরম্যাল করা।
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
            RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Regular + FontStyle.Bold)
        Else 'অন্যথায় এমনিতেই নরম্যাল হবে। নিচের লাইন এর নিশ্চয়তা দেবে।
            RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Regular)
        End If
    ElseIf RichTextBox1.SelectionFont.Italic = False Then 'ইটালিক না থাকলে ইলালিক হবে।
        If RichTextBox1.SelectionFont.Bold = True Then 'বল্ড আছে কি? থাকলে নিচের লাইন নিশ্চিত করবে বল্ডসহ ইটালিক করা।
            RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Italic + FontStyle.Bold)
        Else 'অন্যথায় এমনিতেই ইটালিক হবে। নিচের লাইন এর নিশ্চয়তা দেবে।
            RichTextBox1.SelectionFont = New Font(Me.RichTextBox1.SelectionFont, FontStyle.Italic)
        End If
    End If
    RichTextBox1.Focus() 'ফেকাস হবে টেক্সট বাক্সে।
End Sub
  'ফট পরিবর্তন বোতামের কোড।
Private Sub FontBt_Click(sender As Object, e As EventArgs) Handles FontBt.Click
    Dim fontDialod As New FontDialog 'প্রথম একটি নতুন ফন্ট ডায়লগ ডিফাইন করা চাই।
    fontDialod.Font = RichTextBox1.SelectionFont 'এখন সিলেক্ট করা ফন্ট কি জানতে হবে।
    fontDialod.ShowDialog() 'ফন্ট ডায়লগ প্রদর্শন।
    RichTextBox1.SelectionFont = fontDialod.Font 'সিলেক্ট করা ফন্ট পরিবর্তন।
    RichTextBox1.Focus() 'ফোকাস টেক্সট বাক্সে নিয়ে আসা।
End Sub
'ফট সাইজ বড়ো করার বোতামের কোড। এটা ঠিক নিচের সাবের কোডই- পার্থক্য হলে ৫ পয়েন্ট করে সাইজ বাড়বে।
Private Sub SizeUpBt_Click(sender As Object, e As EventArgs) Handles SizeUpBt.Click
    Try
        RichTextBox1.SelectionFont = New Font(RichTextBox1.SelectionFont.FontFamily, Int
  (RichTextBox1.SelectionFont.SizeInPoints + 5))
    Catch ex As Exception
    End Try
      RichTextBox1.Focus()
End Sub
  'ফট সাইজ ছোট করার বোতামের কোড।
Private Sub SizeDnBut_Click(sender As Object, e As EventArgs) Handles SizeDnBut.Click
    Try 'নিচের দীর্ঘ লাইনটি সিলেক্ট করা ফন্টের সাইজ কমাবে। লক্ষ করুর ৫ পয়েন্ট করে ফন্টের সাইজ কমবে প্রত্যেকবার ক্লিক করলেই।
      RichTextBox1.SelectionFont = New Font(RichTextBox1.SelectionFont.FontFamily, Int
      (RichTextBox1.SelectionFont.SizeInPoints - 5))
      Catch ex As Exception
    End Try
      RichTextBox1.Focus()
End Sub
'ফন্ট রং পরিবর্তন বোতামের কোড। এ কোড নিচের সাবের কোডের অনুরূপ।
Private Sub FontColBt_Click(sender As Object, e As EventArgs) Handles FontColBt.Click
  Dim colorDialog As New ColorDialog
  colorDialog.Color = RichTextBox1.SelectionColor
  colorDialog.ShowDialog()
  RichTextBox1.SelectionColor = colorDialog.Color
  RichTextBox1.Focus()
End Sub
'হাইলাইট রং পরিবর্তন বোতামের কোড।
Private Sub HighLiteColBt_Click(sender As Object, e As EventArgs) Handles HighLiteColBt.Click
  Dim colorDialog As New ColorDialog 'ডিফাইন নতুন একটি কালার ডায়লগ।
  'কোন্ রং আছে তা ডায়লগে দেওয়া।
  colorDialog.Color = RichTextBox1.SelectionBackColor
  colorDialog.ShowDialog() 'ডায়লগ প্রদর্শন
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    'চয়েজকরা রং হবে টেক্সট বাক্সের হাইলাইট রং। নিচের কোডের কাজ এটি।
    RichTextBox1.SelectionBackColor = colorDialog.Color
    RichTextBox1.Focus() 'টেক্সড় বাক্সকে ফোকাসে দাও।
End Sub
'পেছনের রং পরিবর্তন বোতামের কোড।
Private Sub BckColBt_Click(sender As Object, e As EventArgs) Handles BckColBt.Click
    Dim colorDialog As New ColorDialog 'ডিফাইন নতুন একটি কালার ডায়লগ।
    colorDialog.Color = RichTextBox1.BackColor 'কোন্ রং আছে তা ডায়লগে দেওয়া।
    colorDialog.ShowDialog() 'ডায়লগ প্রদর্শন
    RichTextBox1.BackColor = colorDialog.Color 'চয়েজকরা রং হবে টেক্সট্ বাক্সের পেছনের রং।
    RichTextBox1.Focus() 'টেক্সড় বাক্সকে ফোকাসে দাও।
End Sub
'বামদিকে এলাইনের বোতামের কোড।
Private Sub AlinLBt_Click(sender As Object, e As EventArgs) Handles AlinLBt.Click
    RichTextBox1.SelectionAlignment = HorizontalAlignment.Left
End Sub
'মাঝখানে এলাইনের বোতামের কোড।
Private Sub AlinCBt_Click(sender As Object, e As EventArgs) Handles AlinCBt.Click
    RichTextBox1.SelectionAlignment = HorizontalAlignment.Center
End Sub
'ডানদিকে এলাইনের বোতামের কোড।
Private Sub AlnRBt_Click(sender As Object, e As EventArgs) Handles AlnRBt.Click
    RichTextBox1.SelectionAlignment = HorizontalAlignment.Right
End Sub
'কপি বোতামের কোড। লক্ষ করুন, নিচের কাট বোতামের কোড ও এই বোতামের কোড একই-
'তবে একটি লাইন এখানে নেই- মুছে দেওয়ার লাইনটি এখানে প্রযোজ্য নয়।
Private Sub CopyBt_Click(sender As Object, e As EventArgs) Handles CopyBt.Click
    My.Computer.Clipboard.Clear()
    Try
        Clipboard.SetText(RichTextBox1.SelectedText)
    Catch ex As Exception
    End Try
End Sub
'কাট বোতামের কোড।
Private Sub CutBt_Click(sender As Object, e As EventArgs) Handles CutBt.Click
    My.Computer.Clipboard.Clear() 'আগে ক্লিপবোর্ড পরিষ্কার করতে হবে।
    Try 'যাতে সমস্যা হলে প্রোগ্রাম ক্রাশ করে না- তাই এই ছোট্ট কথাটি।
        'প্রথমে ক্লিপর্বোর্ডে সিলেক্ট করা লেখা নিয়ে যেতে হবে- নিচের লাইন এ কাজই করে।
        Clipboard.SetText(RichTextBox1.SelectedText)
            RichTextBox1.SelectedText = "" 'এবার টেক্সট বাক্স থেকে মুছে দিতে হবে লেখাটুকু।
    Catch ex As Exception 'যদি ভুল হয়! ধরো!
    End Try
End Sub
'পেস্ট বোতামের কোড।
Private Sub PstBt_Click(sender As Object, e As EventArgs) Handles PstBt.Click
    If My.Computer.Clipboard.ContainsText Then 'প্রথমে দেখতে হবে ক্লিপবোর্ডে লেখা আছে।
        RichTextBox1.Paste() 'থাকলে ইনসারশন পয়েন্টে পেস্ট করার কোড।
  End If
End Sub
'ফাইল খোলার কোড।
Private Sub OpenFileBt_Click(sender As Object, e As EventArgs) Handles OpenFileBt.Click
    'প্রথমে খোলার ডায়লগ খুলতে হবে।
    Dim openFile1 As New OpenFileDialog()
    'ফাইল আর.এফটি. টাইপ কোথায় আছে দেখতে হবে।
    openFile1.DefaultExt = "*.rtf"
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
        openFile1.Filter = "RTF Files|*.rtf"
        'ফাইল খোলার ডায়লগ দ্বারা কি কোনো ফাইল খোলার নির্দেশ এসেছে?
        If (openFile1.ShowDialog() = System.Windows.Forms.DialogResult.OK) _
            And (openFile1.FileName.Length > 0) Then 'ফাইল নামটি এখন জানা।
          'এখন ফাইলের সবকিছু আমাদের টেক্সট বাক্সে নিয়ে আসতে হবে। এজন্য নিচের কোড লাইনটি।
            RichTextBox1.LoadFile(openFile1.FileName, RichTextBoxStreamType.RichText)
            Me.Text = openFile1.FileName 'এই কোডটুকু দ্বারা প্রদর্শিত ফাইলের প্যাথনেইম ফর্মের উপরে আইকনের পাশে আসবে।
      End If
End Sub
'ফাইল সেইভ করার কোড।
Private Sub SaveFileBt_Click(sender As Object, e As EventArgs) Handles SaveFileBt.Click
        'প্রথমে সেইভ ডায়লগ খুলতে হবে।
    Dim saveFile1 As New SaveFileDialog()
        'ফাইল আর.এফটি. টাইপ হওয়ার জন্য নির্দেশ থাকতে হবে।
        saveFile1.DefaultExt = "*.rtf"
        saveFile1.Filter = "RTF Files|*.rtf"
        'নিচের কোড নিশ্চিত করছে আপনি ফাইল সেইভ ডায়লগে সেইভ করার ইচ্ছা করেছেন। না করলে কিছুই হবে না।
      If (saveFile1.ShowDialog() = System.Windows.Forms.DialogResult.OK) _
          And (saveFile1.FileName.Length > 0) Then 'আপনি ফাইলেন নামও দিয়েছেন বলে নিশ্চিত হওয়া।
          'এবার ফাইল সংরক্ষণ হবে নিচের কোড লাইনটির মাধ্যমে।
            RichTextBox1.SaveFile(saveFile1.FileName, RichTextBoxStreamType.RichText)
      End If
End Sub
'প্রোগ্রাম বন্ধ হওয়ার সময় কিছু লেখা থাকলে এই সাব সক্রিয় হয়ে প্রশ্ন করবে সেইভ করবেন কি না?
Private Sub Form1_Closing(sender As Object, e As System.ComponentModel.CancelEventArgs) Handles
MyBase.Closing
        If RichTextBox1.Text <> "" Then 'বাক্সে লিখা আছে- তা নিশ্চিত করা চাই।
         'নিচের লাইন দ্বারা আপনাকে প্রশ্ন করা ও ফলাফল ভেরিয়েবলে সংরক্ষণ করা হবে।
         Dim Response As String = MsgBox("লেখা সংরক্ষণ করবেন?", vbYesNo)
            If Response = vbYes Then 'যদি ফাইল রক্ষা করতে চান।
                SaveFileBt.PerformClick() 'তাহলে এ সেইভ বোতামে ক্লিক করা হবে স্বয়ংক্রিয়ভাবে।
        End If
        End If
End Sub
 'এই ১৮টি সাবরুটিন হলো আঠারোটি বোতামের মাউস-প্রবেশ সিস্টেম ইভেন্ট সাব।
'আপনি যখন মাউস বোতামের ভেতর নিবেন তখনই বাংলায় কিছু নির্দেশনা দেখবেন।
Private Sub AlinCBt_MouseEnter(sender As Object, e As EventArgs) Handles AlinCBt.MouseEnter
     Dim s As String = "সেন্টার এলাইন" 'প্রথমে একটি স্ট্রিং ভেরিয়েবলে যা লেখা হবে তা সংরক্ষণ করেছি।
     'নিচের লাইনটি হলো  টুলটিপ  নামক সিস্টেম ফাংশন ব্যবহারের লাইন। এর পরেরটি দ্বারা প্রদর্শন হবে।
     Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
     tooltip1.SetToolTip(AlinCBt, s)
End Sub
Private Sub AlinLBt_MouseEnter(sender As Object, e As EventArgs) Handles AlinLBt.MouseEnter
        Dim s As String = "লেফট এলাইন"
        Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
        tooltip1.SetToolTip(AlinLBt, s)
End Sub
Private Sub AlnRBt_MouseEnter(sender As Object, e As EventArgs) Handles AlnRBt.MouseEnter
     Dim s As String = "রাইট এলাইন"
     Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
     tooltip1.SetToolTip(AlnRBt, s)
End Sub
Private Sub BckColBt_MouseEnter(sender As Object, e As EventArgs) Handles BckColBt.MouseEnter
     Dim s As String = "পেছনের রং"
     Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    tooltip1.SetToolTip(BckColBt, s)
End Sub
Private Sub BoldBt_MouseEnter(sender As Object, e As EventArgs) Handles BoldBt.MouseEnter
    Dim s As String = "বল্ড লেখা"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(BoldBt, s)
End Sub
Private Sub Button1_MouseEnter(sender As Object, e As EventArgs) Handles OpenFileBt.MouseEnter
    Dim s As String = "ফাইল খুলুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(OpenFileBt, s)
End Sub
Private Sub Button2_MouseEnter(sender As Object, e As EventArgs) Handles SaveFileBt.MouseEnter
    Dim s As String = "সংরক্ষণ করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(SaveFileBt, s)
End Sub
Private Sub CopyBt_MouseEnter(sender As Object, e As EventArgs) Handles CopyBt.MouseEnter
    Dim s As String = "কপি করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(CopyBt, s)
End Sub
Private Sub CutBt_MouseEnter(sender As Object, e As EventArgs) Handles CutBt.MouseEnter
    Dim s As String = "কেটে দিন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(CutBtn, s)
End Sub
Private Sub FontBt_MouseEnter(sender As Object, e As EventArgs) Handles FontBt.MouseEnter
    Dim s As String = "ফন্ট পরিবর্তন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(FontBt, s)
End Sub
Private Sub FontColBt_MouseEnter(sender As Object, e As EventArgs) Handles FontColBt.MouseEnter
    Dim s As String = "ফন্ট রং"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(FontColBt, s)
End Sub
Private Sub HighLiteColBt_MouseEnter(sender As Object, e As EventArgs) Handles HighLiteColBt.MouseEnter
    Dim s As String = "হাইলাইট রং"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(HighLiteColBt, s)
End Sub
Private Sub ItalicBt_MouseEnter(sender As Object, e As EventArgs) Handles ItalicBt.MouseEnter
    Dim s As String = "ইটালিক লেখা"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(ItalicBt, s)
End Sub
Private Sub PstBt_MouseEnter(sender As Object, e As EventArgs) Handles PstBt.MouseEnter
    Dim s As String = "পেস্ট করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(PstBt, s)
End Sub
Private Sub RedoBt_MouseEnter(sender As Object, e As EventArgs) Handles RedoBt.MouseEnter
    Dim s As String = "আবার করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    tooltip1.SetToolTip(RedoBt, s)
End Sub
Private Sub SizeDnBut_MouseEnter(sender As Object, e As EventArgs) Handles SizeDnBut.MouseEnter
    Dim s As String = "ছোট ফন্ট"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(SizeDnBut, s)
End Sub
Private Sub SizeUpBt_MouseEnter(sender As Object, e As EventArgs) Handles SizeUpBt.MouseEnter
    Dim s As String = "বড়ো ফন্ট"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(SizeUpBt, s)
End Sub
Private Sub UndoBt_MouseEnter(sender As Object, e As EventArgs) Handles UndoBt.MouseEnter
    Dim s As String = "আনডু করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(UndoBt, s)
End Sub
Private Sub OpenTxtBt_MouseEnter(sender As Object, e As EventArgs) Handles OpenTxtBt.MouseEnter
    Dim s As String = "টেক্সট ফাইল খুলুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(OpenTxtBt, s)
End Sub
Private Sub SaveTxtBt_MouseEnter(sender As Object, e As EventArgs) Handles SaveTxtBt.MouseEnter
    Dim s As String = "টেক্সট ফরমাটে সংরক্ষণ করুন"
    Dim tooltip1 As System.Windows.Forms.ToolTip = New System.Windows.Forms.ToolTip()
    tooltip1.SetToolTip(SaveTxtBt, s)
End Sub
'টেক্সট ফাইল খোলার কোড।
Private Sub OpenTxtBt_Click(sender As Object, e As EventArgs) Handles OpenTxtBt.Click
    Dim Open As New OpenFileDialog() 'ফাইল খোলার ডায়লগ ক্লাসের নতুন ইন্সট্যান্ট।
    Dim myStreamReader As System.IO.StreamReader 'একটি সিস্টেম স্ট্রিমরিডার অবজেক্ট ডাইমেনশন করা।
    Open.Filter = "Text[*.txt*]|*.txt|All Files [*,*]|*.*" 'টেক্সট টাইপ ফিল্টার।
    Open.CheckFileExists = True 'খোলার পূর্বে নিশ্চিত করতে হবে ফাইলটি আছে কি না।
    Open.Title = "টেক্সট ফাইল খুলুন" 'ডায়লগের টাইটেল।
    Open.ShowDialog(Me) 'ডায়লগ প্রদর্শন
    Try 'এটা একটি ডিবাগ কিওয়ার্ড। সমস্যা হলে পরের চার লাইন ছেড়ে  ক্যাচ ইএক্স  লাইনে জাম্প দেবে।
        Open.OpenFile() 'চয়েজ করা ফাইল খোলার নির্দেশ।
        myStreamReader = System.IO.File.OpenText(Open.FileName)
        Dim NewRFTText As String = myStreamReader.ReadToEnd() 'ফাইল পুরোটা রিড করার নির্দেশ।
        RichTextBox1.Text = NewRFTText 'ভেরিয়েবলে সংরক্ষিত রিড-করা পুরো ফাইল রিচটেক্সট বাক্সে প্রদর্শন।
      Catch ex As Exception 'কোনো সমস্যা হলো ডায়লগ এসে প্রদর্শন করবে।
    End Try
End Sub
Private Sub SaveTxtBt_Click(sender As Object, e As EventArgs) Handles SaveTxtBt.Click
    If RichTextBox1.Text <> "" Then ' লেখা আছে কি না দেখেই নিচের সাবরুটির কল করতে হবে।
      SaveTheFile() 'নিচের সাবরুটিন কল।
    End If
End Sub
Sub SaveTheFile() 'এই সাবের কাজ হলো সংরক্ষণ ডায়লগ খুলে ফাইন সেভ করা।
    'সঠিকভাবে টেক্সট্ ফরমাটে সংরক্ষণের জন্য রিচ টেক্সট্ বাক্সের লেখা একটি লুকানো টেক্সট্ বাক্সে নিয়ে যেতে হবে।
    TextBox2.Text = RichTextBox1.Text
    Dim Save As New SaveFileDialog() 'ফাইল সংরক্ষণের ডায়লগ ক্লাসের নতুন ইন্সট্যান্ট।
    Dim myStreamWriter As System.IO.StreamWriter 'একটি সিস্টেম স্ট্রিমরিডার অবজেক্ট ডাইমেনশন করা।
    Save.Filter = "Text (*.txt)|*.txt|All files(*.*)|*.*" 'টেক্সট্ টাইপ ফিল্টার।
    Save.CheckPathExists = True 'চয়েজ করা ফাইল সংরক্ষণের প্যাথ সঠিক আছে কি না দেখতে হবে।
```

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

```
    Save.Title = "টেক্সট্ ফরমাটে ফাইল সংরক্ষণ করুন" 'সংরক্ষণ ডায়লগের শিরোনাম।
    Save.ShowDialog(Me) 'এখানে মি অর্থ স্বয়ং ডায়লগটি।
    Try
        'নিচের লাইনটি নিশ্চিত করে একই নামের ফাইল থেকে থাকলে তা প্রতিস্থাপন করা যাবে কি না? না গেলে ফাইল সংরক্ষণ হবে না।
        myStreamWriter = System.IO.File.AppendText(Save.FileName)
        myStreamWriter.Write(TextBox2.Text) 'এবার টেক্সট্ বাক্সে সংরক্ষিত পুরো টেক্সট্ ফাইলে লেখা হবে।
        myStreamWriter.Flush() 'মেমোরি থেকে ক্লাসটি মুছে দিতে হবে।
      Catch ex As Exception
    End Try
  End Sub
End Class
```

কোডের ব্যাখ্যঃ প্রথমেই লক্ষ করুন, বাংলা ইউনিকোডে প্রচুর কমেন্ট দিয়েছি। মাইক্রসফটকে ধন্যবাদ! তারা অপারেটিং সিস্টেমে বাংলায় উইনিকোড ক্যারাক্টার যুক্ত করে আমাদের বিরাট উপকার করেছেন। সুতরাং ভিজুয়েল বেসিক.নেট এর আইডিইতে আমরা অনায়াসেই বাংলা ইউনিকোডে লেখার সুযোগ পেয়েছি। প্রোগ্রামেও বাংলায় নির্দেশনা পর্যন্ত প্রদর্শন হওয়ার ব্যবস্থা হয়েছে। যদিও এখন পর্যন্ত রিচটেক্সটবক্স কন্ট্রোলের মধ্যে ইউনিকোড টাইপ করার সুযোগ তৈরি করে দেওয়া হয় নি। তবে আমরা আশাবাদী তা অচিরেই হয়ে যাবে।

যাক ফিরে আসি কোডের ব্যাখ্যায়। নিচে একে একে প্রতিটি সাবরুটিনের ব্যাখ্যা লাইন বাই লাইল তুলে ধরছি। প্রয়োজনে আলোচিত সাবরুটিন পুনরায় টাইপ করে দিয়েছি। লক্ষ করুন, সকল কোডই ফর্মের ক্লাস সাবরুটিনের ভেতর। আমরা কোনো আলাদা মডিউল বা ক্লাস তৈরি করি নি।

১. প্রথমেই একটি গ্লোবাল ইন্টেজার ভেরিয়েবল ডিক্লায়ার করা হয়েছে। এটি দ্বারাই আমরা প্রোগ্রাম চলাকালে রিচ টেক্সট্ বাক্সের সাইজ সঠিক রাখবো। কারণ ব্যবহারকারী প্রয়োজনে ফর্মের সাইজ রদবদল করতে পারেন।
২. `Form1_Load` সাবের কোডঃ এটির দু'টি কাজ আছে: (ক) রিচ টেক্সট্ বাক্সের সাইজ সেট করা ও (খ) চিচ টেক্সট্ বাক্সে সাউস রাইট-ক্লিককালে একটি কনটেক্সট্ মেনু প্রদর্শন করা। কনটেক্সট্ মেনুর মধ্যে কমি-পেস্ট-কাট- এ তিনটি সুবিধা থাকবে। ব্যবহারকারী লেখা বা লেখাংশ সিলেক্ট করে রাইট-ক্লিকের মাধ্যমে কনটেক্সট্ মেনু চয়েজ করে সহজেই এ তিনটি কাজের যে কোনো একটি করতে সক্ষম হবেন। সুতরাং সাবের প্রথম লাইন দ্বারা রিচ টেক্সট্ বাক্সের উচ্চতা ও প্রস্থ নির্ধারিত করে দেওয়া হয়েছে। দ্বিতীয় লাইনের কাজ হলো একে ফর্মের কেন্দ্রে নিয়ে আসা। আর তৃতীয় লাইন দ্বারা পরের সাবরুটিনটি কল করা হয়েছে যার কাজ হলো কনটেস্টট্ মেনু তৈরি করা।
৩. `InitializeMyContextMenu()` সাবের কোডঃ আগেই বলেছি, এটা আমাদের সৃষ্ট কনটেক্সট্ মেনু তৈরির সাবরুটিন। প্রথমে ডিফাইন করতে হবে একটি নতুন কনটেক্সট্ মেনু আইটেম অবজেক্ট। এরপর ডিফাইন করতে হবে তিনটি মেনু আইটেম নাম ও প্রতিটির সঙ্গে সম্পৃক্ত বোতামের একশন। তিন জোড়া কোড লাইনে এ কাজগুলো করা হয়েছে। সবশেষে প্রতিটি মেনু আইটেম দেখাতে হবে যখনই মাউসে রাইট-ক্লিক করা হবে এবং পুরো মেনুর সঙ্গে সমপৃক্ত করতে হবে রিচ টেক্সট্ বাক্সটি।
৪. `Create_Resize` নামক ফর্মের ইভেন্ট সাবের কোডঃ প্রথমত এই সাবরুটিন স্বয়ংক্রিভাবে তৈরি করতে হলে কোড লিখার ফিল্ডের উপরস্থ বায়ের কম্বো বাক্সে যেয়ে চয়েজ করুন ফর্ম [Form1 Events]। এবার ডানের কম্বো বাক্সে যেয়ে চয়েজ করুন রিসাইজ [Resize]। সাথে সাথেই এই সাবরুনিটি আসবে। এর ভেতরেই কোড লিখতে হবে। প্রথম ইফ-এন্ড ইফ [If - End If] ব্লকের কাজ হলো ফর্মের সাইজ পরিবর্তন হলে রিচ টেক্সবাক্সের সাইজও পরিবর্তন করা। চারটি লাইনের প্রথমটি দ্বারা ফর্মের প্রস্থের সাথে সামঞ্জস্য রেখে রিচ টেক্সট্ বাক্সোর প্রস্থও পরিবর্তন করা হয়েছে। উভয় দিকে ১০০ পন্টে বর্ডারও রাখা হয়েছে। শেষের লাইন দ্বারা নিশ্চিত করা হয়েছে ফর্মের মাঝখানে যেনো রিচ টেক্সট্ বাক্স থাকে। সাবের দ্বিতীয় ইফ-এন্ড ইফ [If - End If] অনুরূপ কাজ করে। তবে এখানে প্রস্থ বাড়ছে না বরং কমেছে- এটাই পার্থক্য। লক্ষ করুন, তার ক্লাসের ফেতর ফর্মকে ইঙ্গিত করতে হলে Me কিওয়ার্ড ব্যবহৃর করা দরকার।
৪. `Create_SizeChanged` নামক ফর্মের ইভেন্ট সাবের কোডঃ এই কোডের কোড ও ক্রিয়া উপরে আলোচিত সাবের

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

মতোই। সুতরাং, শিক্ষার্থীতা একে ভালো করে দেখে নেবেন ও বাংলায় লেখা কমেন্ট পাঠ করবেন। আশাকরি বুঝতে অসুবিধা হবে না।

৫. UndoBt_Click ও RedoBt_Click নামক বোতামের ক্লিক একশন ইভেন্ট সাব: এই উভয় সাব একইভাবে কাজ করে। সাব দু'টি যথাক্রমে Undo() এবং Redo() নামক সিস্টেম ফাংশনের মাধ্যমে ক্রিয়া করে।

৬. BoldBt_Click ও ItalicBt_Click নামক বোতাম দু'টোর ক্লিক একশন ইভেন্ট সাব: এ দু'টোর কাজও অনুরূপ। সুতরাং কিছুটা ব্যাখ্যা দ্বারা উভয়টির ক্রিয়া বুঝা যাবে। উভয় কোডে ব্যবহৃত হয়েছে একটি If - ElseIf - End If ব্লক। উভয়টির মাঝে আবার আরেকটি Else - End If ব্লক আছে। দেখতেই পাচ্ছেন প্রথম ইফের কাজ হলো বল্ড থেকে থাকলে নরম্যাল করা- তবে দেখতে হবে সিলেকশন ইটালিকও আছে কি না- থাকলে ইটালিকসহ নরম্যাল করা চাই। অনুরূপ যদি বল্ড না থাকে তাহলে ইটালিক থাকলে সেটাসহ বল্ড বানাতে হবে। ElseIf অংশের কাজ এটি। আশারাখি শিক্ষার্থীরা লজিকটি বুঝে নিয়েছেন।

৭. FontBt_Click বোতামের ক্লিক একশন ইভেন্ট সাব: এটার কাজ ফন্ট পরিবর্তন করা। ভিজুয়্যাল বেসিক.নেট ফন্টলিস্টসহ একটি জানালা তৈরি করে দিয়েছে। সেখান থেকে সিস্টেমে ইন্সটল করা সকল ফন্টের মধ্যে যে কোনো একটি চয়েজ করা যায়। সিলেক্ট করা টেক্সট কিংবা ইনসারশন পয়েন্ট থেকে নতুন ফন্ট কাজ করবে। কোডের কাজ হলো ডায়লগটি খোলা ও সিলেক্টকরা ফন্ট পরিবর্তন করা।

৮. SizeUpBt_Click ও FontColBt_Click নামক বোতাম দু'টোর ক্লিক একশন ইভেন্ট সাব: এ দু'টো বোতামের সাব একইভাবে কাজ করে। প্রথমটি ৫ পয়েন্ট করে সিলেক্ট করা ফন্টের সাইজ বাড়াবে ও দ্বিতীয়টি ৫ পয়েন্ট করে কমাবে। এই যা। লক্ষ করুন সাইজ বাড়ানো ও কমানোর জন্য ব্যবহৃত হয়েছে দীর্ঘ এই একটিমাত্র কোড লাইন:

```
RichTextBox1.SelectionFont = New Font(RichTextBox1.SelectionFont.FontFamily, Int(RichTextBox1.SelectionFont.SizeInPoints - 5))
```

লাইনটির = সাইনের ডানের অংশ হলো একটি সিস্টেম ফাংশন। এর ভেতর প্যারামিটার হিসাবে আছে ব্যবহৃত ফন্ট ফ্যামিলি ও ইন্টেজার টাইপের সাইজ।

৯. FontColBt_Click নামক বোতামের ক্লিক একশন ইভেন্ট সাব: এ সাবটি উপরের (৭নং) ফন্ট পরিবর্তন সাবের মতোই কাজ করে। এ ক্ষেত্রে ডায়লগটি হলো কালার ডায়লগ। এই যা। পরের দু'টি বোতামের একশন ইভেন্ট কোডও কালার ডায়লগ খুলে যথাক্রমে সিলেক্ট করা লেখার হাইলাইট রং ও রিচ টেক্সট বাক্সের বেকগ্রাউন্ড রং ব্যবহারকারীর ইচ্ছেমাফিক পরিবর্ত করবে। সুতরাং এগুলো সম্পর্কে আর বেশি কিছু বলার নেই।

১০. AlinLBt, AlinCbt এবং AlinRbt নামক বোতাম তিনটির ক্লিক একশন ইভেন্ট সাব: এগুলোর কাজ কি তা-তো সকলেই বুঝতে পারছেন। তিনটি ক্ষেত্রেই কাঙ্ক্ষিত ফলাফল পেতে ব্যবহার করেছি HorizontalAlignment নামক সিস্টেম ফাংশন।

১১. CopyBt, CutBt এবং PstBt বোতাম তিনটির ক্লিক একশন ইভেন্ট সাব: এ তিনটি বোতামের কারণও আমাদের জানা। তবে লক্ষ করুন, প্রথম দু'টিতে Try - End Try ব্লকের ভেতর কোড লিখা হয়েছে। এর কারণ হলো ক্লিপ বোর্ডে সবকিছু যেতে না-ও পারে। কপি এবং কাট- উভয় বোতামের কাজ হলো ক্লিপবোর্ডে সিলেকটেড টেক্সট যথাক্রমে কপি বা মুছে কেটে নেওয়া। সিলেকশন যদি না থাকে- কিংবা টেক্সট ছাড়া অন্য কিছু থাকে তাহলে যে কোডিং সমস্যা হবে সেটা এড়াতে উক্ত Try - End Try ব্লক ব্যবহৃত হয়েছে। কপি করতে যেয়ে আমরা সিস্টেমের Clipboard নামক ফাংশনটি ব্যবহার করেছি। এর প্যারামিটার হলো সিলেকটেড টেক্সট। লক্ষ করুন দ্বিতীয় বোতামের কোডে অতিরিক্ত একটি লাইন দ্বারা সিলেকটেড টেক্সট মুছে দেওয়ার নির্দেশনা দিয়েছি। তৃতীয় বোতামের কোড কিছুটা ভিন্ন। সেখানে একটি If - End If ব্লকের ভেতর সিস্টেমের Paste কিওয়াডটি ব্যবহার করেছি। ক্লিপবোর্ডে কিছু আছে কি না তা দেখেই পেস্ট করতে হবে।

১২. OpenFileBt ও SaveFileBt বোতামদ্বয়ের ক্লিক একশন ইভেন্ট সাব: এ দু'টোর একশন প্রায় একই। প্রথমটি যে ডায়লগ খুলে তাহলো OpenFileDialog() ডায়লগ। আর দ্বিতীয়টি খুলে SaveFileDialog() ডায়লগটি। উভয়টিই সিস্টেমের ফাংশন। লক্ষ করুন, ডায়লগ খোলার জন্য নতুন ডায়লগ ডিক্লারেশন পরে আমরা টাইপ এক্সটেনশন ও ফিল্টার নির্দিষ্ট করেছি। আমাদের উদ্দেশ্য শুধুমাত্র .rtf ফাইল নিয়ে কাজ করা। এটাই হচ্ছে রিচ টেক্সট বাক্সের ফাইল টাইপ। প্রথম সাাবে আমরা ইফ - এন্ড ইফ টেস্টের মাধ্যমে নিশ্চিত করেছি যে ব্যবহারকারী ফাইল খুলেছেন ও ফাইলটি শূন্য নয়। এরপর একটি মাত্র লাইনে নির্দেশ দেওয়া হয়েছে ফাইল খুলে সব লেখা রিচ টেক্সট বাক্সে প্রদর্শন করতে। এর

১৪তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

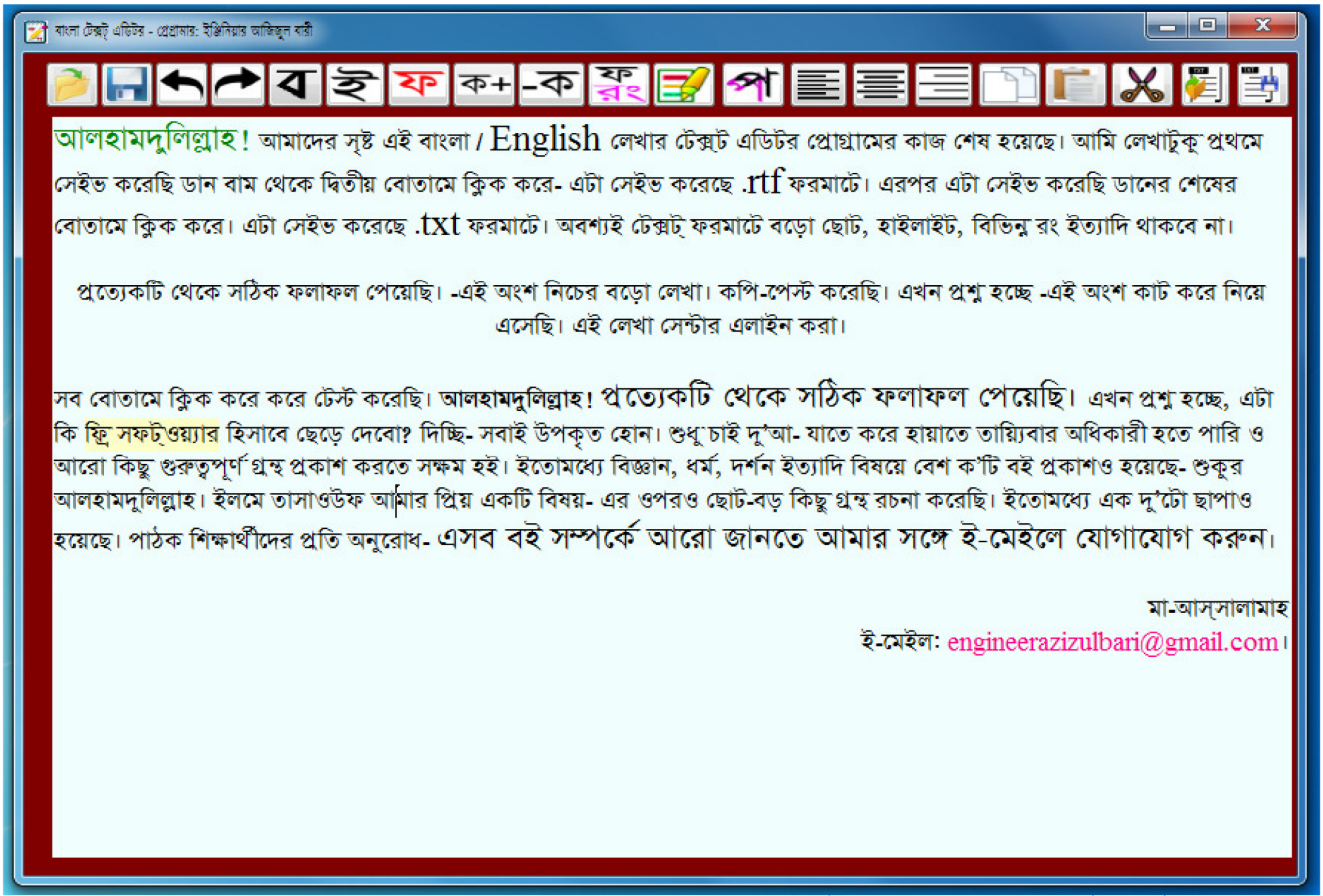
পরের লাইন ফাইলের প্যাথ নাম জানালার আইকনের পাশে প্রদর্শন করবে।
দ্বিতীয় সাবের কাজও অনুরূপ। তবে লক্ষ করুন, সেইভ করা ফাইল কোথায় সংরক্ষিত হয়েছে সে প্যাথ নাম জানালার আইকনের পাশে লিখার নির্দেন দেই নি। তবে চাইলে দেওয়া যেতো।
১৩. Form1_Closing ইভেন্ট সাবের কোড: কম্পিউটারের প্রতিটি কাজ সম্পাদনে একটি ঘটনা বা ইভেন্ট সিস্টেম সফটওয়্যার থেকে আসে। সফটওয়্যার ডেভেলাপার হিসেবে আমাদের কাজ হলো এসব ইভেন্টকে ধরে ইচ্ছেমতো কাজে লাগানো। ইভেন্ট জেনারেট হয় কিবোর্ড, মাউস ইত্যাদি থেকে। ফর্ম বন্ধ করার সময় আমরা মাউস দ্বারা একটি আইকনে [জানালার ডান পাশের x-ক্রসে] ক্লিক করি। এতে একটি সিস্টেম ইভেন্ট সক্রিয় হয়। এটির নামই Form1_Closing ইভেন্ট। আমরা একটি সাবে একে ধরে কিছু নির্দেশনা দিয়েছি। একটি ডবল নেস্টেড্ ইফ - এন্ড ইফ ব্লকে এই কোড লিখেছি। আমাদের উদ্দেশ্য হলো, রিচ টেক্সট বাক্সে কিছু লেখা থাকলে তা সংরক্ষণ করা হবে কি না সেটা ব্যবহারকারীকে প্রশ্ন করা। জবাব 'হ্যাঁ' হলে সংরক্ষণ করে প্রোগ্রাম ক্লোজ করা- অন্যথায় এমনিতেই ক্লোজ করা। লক্ষ করুন, সেইভ করার জন্য আমরা পুনরায় সেইভ ডায়লগ খোলার কোড এখানে লিখি নি- বরং সেইভ বোতামে স্বয়ংক্রিয়ভাবে ক্লিক ইভেন্ট জেনারেট করার নির্দেশ দিয়েছি। সাবের কোড রান করা শেষ হয়ে গেলেই কিন্তু প্রোগ্রাম বন্ধ হয়ে যাবে।
১৪. আমাদের ২০টি বোতামের ওপর ব্যবহারকারী মাউস নিয়ে গেলে বাংলায় কিছু তথ্যমূলক লেখা দেখতে পাবেন। অর্থাৎ কোন্ বোতামের কাজ কোনটি তা জানতে পারবেন। যে উপায়ে এটা জানানো হয় তাকে বলে SetToolTip। এটা সিস্টেমের একটি ফাংশন যার প্যারামিটার হলো অবজেক্ট বা বস্তুর নাম [এ ক্ষেত্রে বোতামের] ও পপাপ বাক্সে কি লিখা থাকবে সে স্ট্রিং। অর্থাৎ যদি এভাবে ফাংশন ব্যবহার করি: `SetToolTip (Button1, "□ □□□□□□□□□□□ □□□□□")` তাহলে ঐ বোতামের ওপর মাউস নিলেই ছোট্ট একটি পপাপ জানালা আসবে যেখানে লিখা থাকবে, □ □□□□□□□□□□□ □□ □□□□ মোটকথা দ্বিতীয় প্যারামিটারের ভেতর যা-ই লিখা থাকুক না কেনো, তা জানালায় প্রদর্শিত হবে। আমরা ২০টি বোতামের টুল টিপ লিখে দিয়েছি যাতেকরে ব্যবহারকারী জানতে পারেন কোন্ বোতামে ক্লিক করলে কোন্ একশন হবে।

১৫. `OpenTxtBt`, `SaveTxtBt` ও `SaveTheFile()` নামক ফাংশন সাবের ব্যাখ্যা: প্রথম দু'টি হলো বোতামের ক্লিক ইভেন্ট সাব ও শেষেরটি আমাদের সৃষ্ট একটি ফাংশন। আমরা চাই ব্যবহারকারী প্রয়োজনে লেখাটুকু টেক্সট ফরমাটে সেইভ করা ও একই ফরমাটের লেখা রিচ টেক্সট বাক্সে নিয়ে আসার সুযোগ পাবেন। এই তিনটি সাবের কাজ তা-ই। প্রথমটি ও শেষেরটি মূলত আগে ব্যাখ্যাত OpenFileBt ও SaveFileBt বোতামদ্বয়ের সাবের অনুরূপ। এখানে পার্থক্য হলো টাইপ ও ফিল্টার .rtf এর বদলে .txt লিখেছি। তবে লক্ষ করুন, সংরক্ষণ নির্দেশের পূর্বে পুরো লেখাটুকু একটি টেক্সট ফরমাটের টেক্সট বাক্সে নিয়ে আসা হয়েছে। এই বাক্সটি কিন্তু লুকানো। এতে নিয়ে আসার ফলে টেক্সট ফরমেটিং সঠিক হবে। যাক, কোডের ব্যাখ্যা এটুকুই- যেহেতু কোডের মধ্যেই বাংলায় ব্যাখ্যামূলক কমেন্ট করেছি তাই আর অতিরিক্ত বলার প্রয়োজন নেই। আমি আশারাখি শিক্ষার্থীরা কোড বুঝতে খুব একটা সমস্যায় পড়বেন না। এবার দেখে নিই প্রোগ্রমাটি একবার রান করে- কেমন? পরের পৃষ্ঠার চিত্রটি দেখুন। ভেতরে কি লেখা আছে তা-ও দয়া করে পাঠ করবেন!

১৬তম পরিচ্ছেদ: *Complete Application Tutorials*

পূর্ণাঙ্গ এ্যাপলিকেশন প্রশিক্ষণ : *Complete Application Tutorials*

আমরা মোট ৪টি ব্যবহারযোগ্য মোটামুটি জটিল এ্যাপলিকেশন তৈরি করতে সক্ষম হয়েছি। আরো দু'টি প্রোগ্রাম ডেভেলাপ করবো শিক্ষার্থীদের সুবিধার জন্য। তবে এক্ষুণি নয়। পরের পৃষ্ঠাসমূহে ভিজুয়্যাল বেসিক.নেট -এর একটি শক্তিশালী ডেভেলাপমেন্ট টুল সম্পর্কে কিছুটা আলোকপাত করার প্রয়াস পাচ্ছি। চাক্ষুষ্মান শিক্ষার্থীদের মনে হয়তো প্রশ্ন জেগেছে, অনেক গুরুত্বপূর্ণ কম্পিউটার সফ্‌টওয়্যার সিস্টেম 'ডটাবেইজ' সম্পর্কে এই নীরবতা কেনো? এ



প্রশ্নের জবাবেই পরের পুরো অনুচ্ছেদটির অবতারণা। ডাটাবেইজ করার জন্য পর্যাপ্ত টুলস্ ও ফাংশন অবশ্যই .নেট-এ আছে। আমরা অবশ্যই এডভান্স ডাটাবেইজ প্রোগ্রামিং সম্পর্কে কিছু বলবো না। এটা এই গ্রন্থের স্কোপের বাইরে। তবে বেসিক বিষয়গুলো অবশ্যই জেনে নেবেন। আমরা শেষের দিকে একটি বাংলা ঠিকানাবই ডাটাবেইস তৈরিও করবো। শিক্ষার্থীরা মৌলিক বিষয়গুলো রপ্ত করে নিতে পারলে, ধীরে ধীরে এডভান্স ডাটাবেইজও ডেভেলাপ করতে পারবেন নিশ্চয়ই। কারণ এটকু পরই দেখতে পাবেন, মাইক্রসফটের 'Access' ডাটাবেইজের টেবিল কিভাবে অতি

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

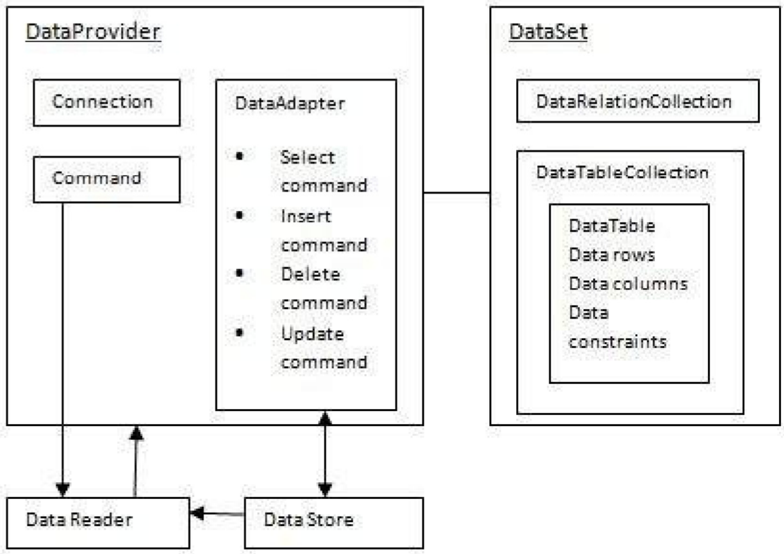
সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

অনেকেই ভাবেন, ডাটাবেইজ প্রোগ্রামিং বুঝি খব জটিল-কঠিন। বাস্তবে তা মোটেই নয়। অন্তত, ভিজুয়্যাল বেসিক.নেট রিলিজ হওয়ার পর থেকে এই ধারণা যে পাল্টে গেছে তা নিশ্চিত করে বলা যায়। এ পরিচ্ছেদ থেকে বিষয়টি অবশ্যই শিক্ষার্থীদের নিকট স্পষ্ট হবে। বাস্তবে সফ্টওয়্যার ডেভেলাপমেন্ট মূলত একটি চিত্তাকর্ষক ক্রিয়ার নাম। তবে যা প্রয়োজন তাহলো, শিক্ষাগ্রহ ও কিছুটা সময়। যাক, এবার আমরা একে একে ডাটাবেইজ প্রোগ্রামিংয়ের ওপর উদাহরণসহ বিস্তারিত আলোচনা করতে চাচ্ছি।

১. এক্টিভএক্স ডাটা অবজেক্ট.নেট [সংক্ষেপে: ADO.Net]

এ্যাপলিকেশন দু'টি উপায়ে সংরক্ষিত ডাটা বা তথ্য নিয়ে কাজ করতে পারে: ১. ব্যবহার-বান্ধব উপায়ে প্রদর্শনের জন্য সংরক্ষিত ডাটা নিয়ে আসা। ২. সংরক্ষিত ডাটাকে প্রয়োজনে পরিবর্তন, পরিবর্ধন, সংযোজন ও মুছে দেওয়া এবং পুনরায় সংরক্ষণ নিশ্চিত করা। মাইক্রসফট .নেট সৃষ্টিতে এডো.নেট [ADO.Net] নামক একটি মডেল তৈরি করেছে। এটা মূলত .নেট এর ফ্রেমওয়ার্কের একটি অংশ। ফলে .নেট কোনো ডাটাসূত্র যেমন মাইক্রসফটের Access থেকে ডাটা সহজেই নিয়ে এসে কাজ করতে পারে।

(ক) ADO.Net Object Model - এডো.নেট অবজেক্ট মডেল: এটা মূলত কাঠামোগত একটি প্রসেসের নাম। ডাটাবেইজের সঙ্গে সম্পৃক্ত বিভিন্ন উপাদানে কিভাবে তথ্য চলাফেরা করে তারই একটি উপায়-অবলম্বন। আমরা একে চিত্রের মাধ্যমে তুলে ধরতে পারি। যেমন:



লক্ষ করুন, ডাটা কোনো ডাটাবেইজ বা স্টোরে সংরক্ষিত থাকে। এটা DataProvider দ্বারা বের করে নিয়ে আসা হয়। এই প্রভাইডারের ভেতর বেশ ক'টি অংশ আছে যেগুলোর কাজ হলো ডাটা এ্যাপলিকেশনে নিয়ে আসা যাতেকরে আপডেট করা যায়।

এ্যাপলিকেশনে ডাটা গ্রহণ করে হয় DataSet না হয় DataReader এর মাধ্যমে।

* DataSet তথ্য সংরক্ষণ করে রাখে একটি আলাদা ক্যাশিতে। সেটা এ্যাপলিকেশন নিয়ে আসে।

* DataReader এ্যাপলিকেশনে ডাটা সংযোজন করে শুধুমাত্র পাঠ [read-only] ও অগ্রগামী [forward-only] তথ্য হিসাবে।

(খ) DataProvider - তথ্য সরবরাহকারী : এটা ব্যবহৃত হয় ডাটাবেইজের সঙ্গে সংযুক্তি, কমান্ড এক্সেকিউট, তথ্য আদান-প্রদান, ডাটাসেটে সংরক্ষণ, পড়া ও পরিবর্তন ইত্যাদি কাজের জন্য। ADO.Net -এ ডাটা প্রভাইডারে আছে চারটি অবজেক্ট। যথা:-

(১) Connection - সংযুক্তি: বুঝাই যাচ্ছে এ অংশের কাজ হলো কোনো ডাটা সূত্রের সঙ্গে সংযুক্তি সৃষ্টি করা।

(২) Command - নির্দেশ: কমান্ড হলো কোনো SQL [Structured Query Language] স্টেটমেন্ট কিংবা স্টোরকরা প্রসিডোর। এরই মাধ্যমে ডাটা সূত্রের তথ্যাদি পরিবর্তন, মুছা, সংরক্ষণ ও সংযোজন ইত্যাদি কাজ করা হয়।

(৩) DataReader - তথ্য পাঠক: বুঝাই যাচ্ছে এ অংশের কাজ হলো কোনো ডাটা সূত্র থেকে শুধুমাত্র পাঠ ও

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

অগ্রগামী মোডে কাজ করা। তথ্যাদি পরিবর্তন করা যাবে না।

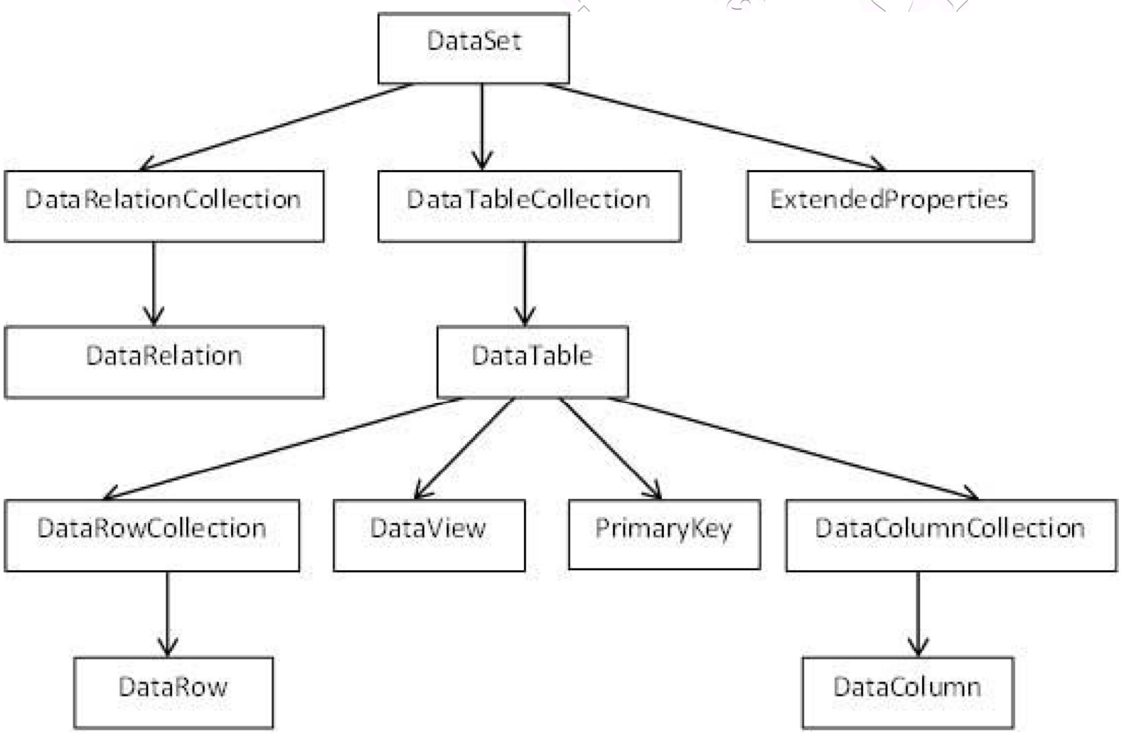
(৪) DataAdapter - তথ্য অভিযোজক: এটা ADO.Net এর একটি গুরুত্বপূর্ণ বস্তু। এই তথ্য অভিযোজকের মাধ্যমেই যাতবীয় ডাটা ডাটাবেইজ থেকে আদান-প্রদান হয়। এটা ডাটা ডাটাবেইজ থেকে ডাটাসেটে নিয়ে আসে এবং এটাই আবার ডাটা আপডেট করে। ডাটাসেটে রদবদল হলে এই ডাটাএডাপটারই ডাটাবেইজে পরিবর্তন আনয়ন ও সংরক্ষণ করে।

ADO.Net -এ বেশ ক'টি ডাটা সরবরাহকারী আছে। যেমন:

* SQL সার্ভারের জন্য .Net Framework ডাটা সরবরাহকারী।
* OLE [Object Linking and Embedding] ডাটাবেইজের জন্য .Net Framework ডাটা সরবরাহকারী।
* ODBC [Open Database Connectivity] ডাটাবেইজের জন্য .Net Framework ডাটা সরবরাহকারী।
* Oracle [Oak Ridge Automatic *Computer* and Logical Engine] ডাটাবেইজের জন্য .Net Framework ডাটা সরবরাহকারী।
* EntityClient সরবরাহকারী- এর মাধ্যমে EDM [Entity Data Model] এ্যাপলিকেশন ব্যাবহার করে ডাটা সংগ্রহের ব্যবস্থা।

DataSet

এটা হলো মেমোরির -মধ্যে তথ্য উপস্থাপনের একটি উপায়। এটা অসংযুক্ত, ক্যাশিতে সংরক্ষিত রেকর্ড সেট যা কোনো ডাটাবেইজ থেকে প্রাপ্ত হওয়া যায়। যখনই কোনো ডাটাবেইজের সঙ্গে সংযুক্তি সৃষ্টি হয় তখন, ডাটা এডাপটারে একটি ডাটাসেট তৈরি করে এতে সংরক্ষণ করে। অপরদিকে ডাটাবেইজের সঙ্গে সংযুক্তি বন্ধ হয়ে যায়। এটাকে বলে 'disconnected architecture'। ডাটাসেট একটি Virtual ডাটাবেইজ হিসেবে কাজ করে। এতে থাকে টেবিল, রো এবং কলাম। নিচের চিত্র থেকে হয়তো এই ডাটাসেট অবজেক্ট মডেলটি সম্পর্কে আরো বেশি অবগতি সম্ভব হবে।



১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

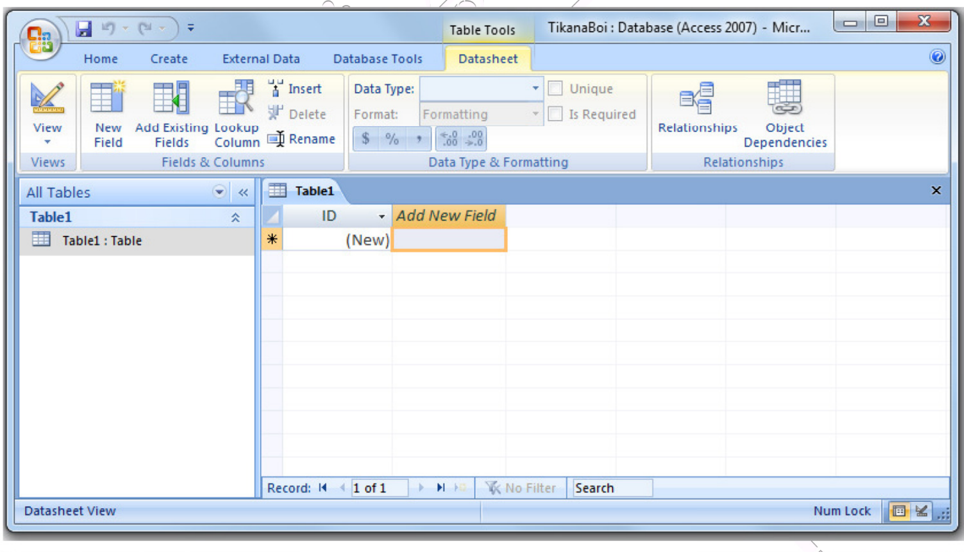
যাক, খুব বেশি টেকনিক্যাল কথা নিয়ে মাথা ঘামাবার প্রয়োজন নেই- উদাহরণের মাধ্যমে যে কোনো বিষয় সম্পর্কে জ্ঞানার্জন একটি উত্তম উপায়। এছাড়া ডাটাবেইজের মতো একটি উচ্চ পর্যায়ের কম্পিউটার প্রোগ্রামিং সম্পর্কে এ গ্রন্থে খুব একটা বেশি বলাও সম্ভব নয়। এ বিষয়ের উপর আলাদাভাবে অনেক গ্রন্থাদি রচিত হয়েছে। আমাদের উদ্দেশ্য হলো, ভিজুয়্যাল বেসিক.নেট ও ডাটাবেইজ এ দু'য়ের মধ্যে তথ্য আদান-প্রদান সম্পর্কে মৌলিক কিছু জ্ঞানার্জন- এই যা। সুতরাং আসুন, প্রথমে একটি উদাহরণ প্রজেক্ট তৈরি করি।

১. ভিজুয়্যাল বেসিক.নেট ও ডাটাবেইজ: একটি ঠিকানা বই ডাটাবেইজ প্রোগ্রাম

ঠিকানা বই ডাটাবেইজ প্রোগ্রামের জন্য মাইক্রসফটের Access ২০০৭ কিংবা এর পরের কোনো ভার্সন আপনার কম্পিউটারে থাকতে হবে। নিম্নের পদক্ষেপগুলো অনুসরণ করে ডাটাবেইজ প্রোগ্রামটি তৈরি করুন।

১. প্রোগ্রাম ফাইলে যেয়ে Access খুলুন। একটি নতুন ডাটাবেইজ তৈরি করুন। এর নামকরণ করুন 'TikanaBoi'। যেখানে ইচ্ছা এটা সেইভ করুন। তবে মনে রাখবেন, পরবর্তিতে এই ফাইলের প্যাথনেইম আমাদের দরকার।

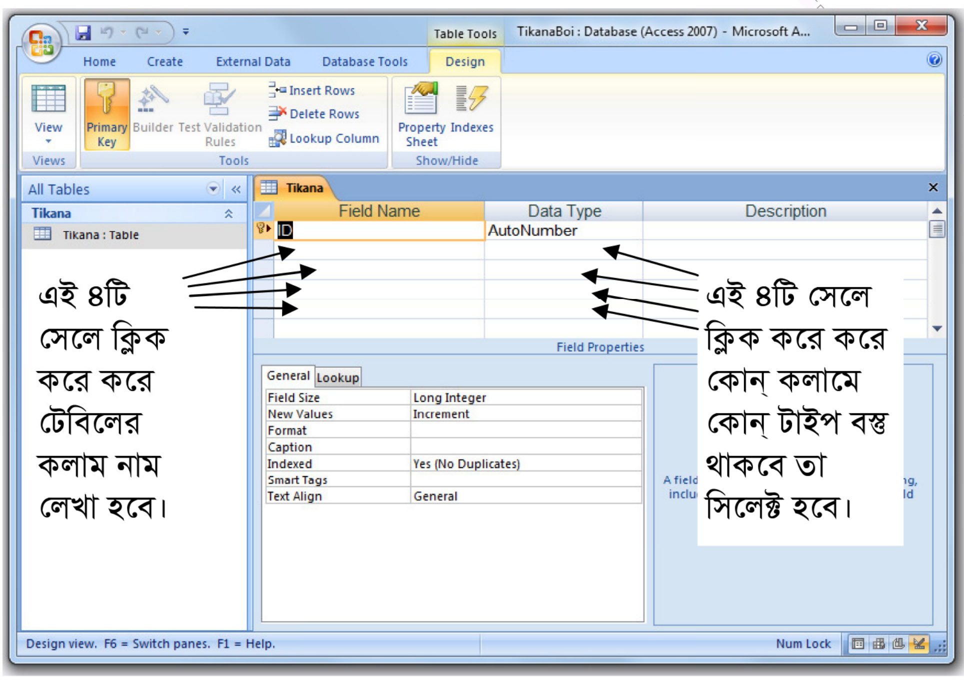
২. প্রথমেই আপনি নিচের চিত্র মুতাবিক জানালাটা দেখতে পাবেন।

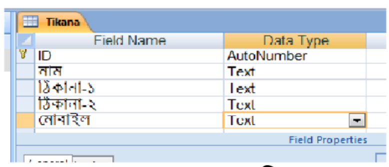


৩. বায়ের উপরের কোণে  ডানে চিত্রিত আইকনে ক্লিক করুন। যে জানালা আসবে তাতে টেবিলের নাম দিন: 'Tikana'।

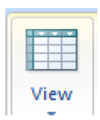
৪. এবার যে জানালাটি আসবে তার একটি চিত্র নিচে দেখানো হয়েছে।

৫. কোন্ কোন্ সেলে আমরা লিখবো তা চিত্রে দেখানো হয়েছে এ্যারো দ্বারা। Field Name কলামের দ্বিতীয় আইটেম হবে: 'নাম', পরেরটি হবে 'ঠিকানা-১', এরপর হবে 'ঠিকানা-২' এবং সবশেষের সেলে হবে 'মোবাইল'। Data Type কলামের চারটি সেলেই 'Text' শব্দটি থাকবে। দেখুন নিচের চিত্রটি। ব্যস্ ডিজাইন জানালার কাজ



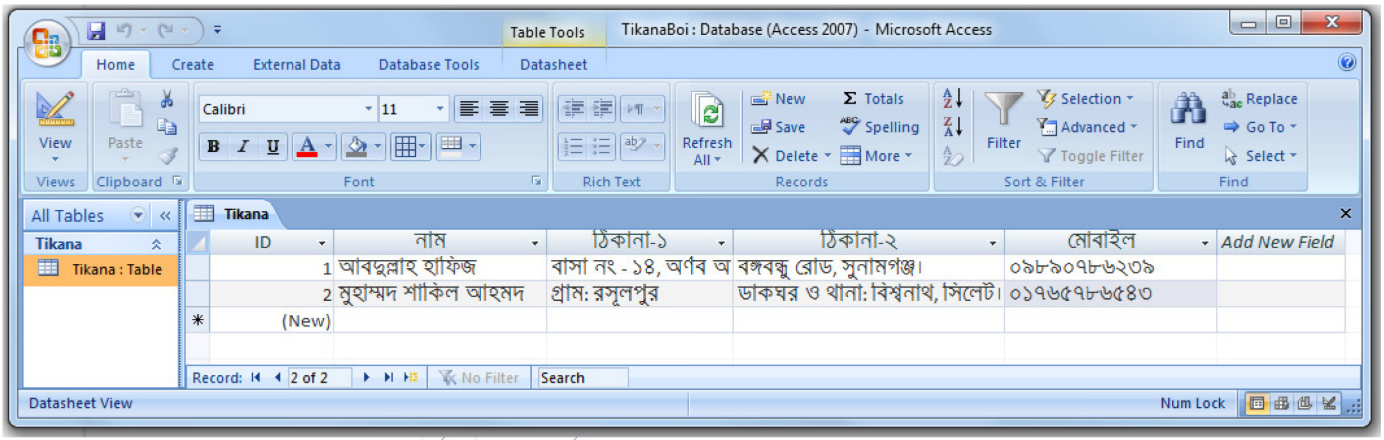


শেষ হলো। এবার ক্লিক করুন উপরের কোণায় স্থাপিত নিচে চিত্রিত আইকনে। একটি ডায়লগ আসবে সেইভ করার জন্য। এরপর আসবে টেবিল ভিউ, যেখানো আমরা তথ্য লিখতে পারি।



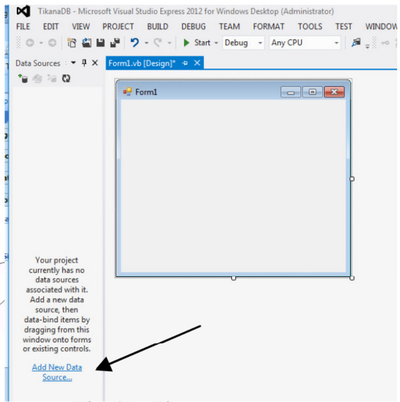
সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

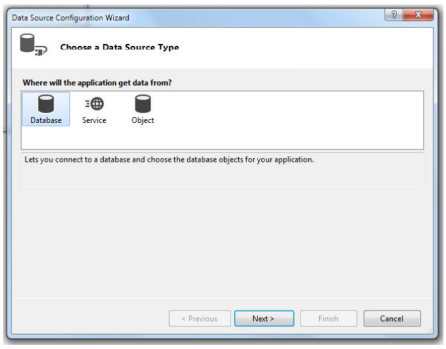
৬. প্রতিটি কলামের প্রতিটি সেলে বাংলা ইউনিকোডে লিখুন। কি লিখবেন ও কোথায় লিখবেন তা নিচের চিত্রে দেওয়া হয়েছে। অবশ্য আপনি চাইলে নাম, ঠিকানা ইত্যাদি ভিন্নও লিখতে পারেন। সেইভ করে বন্ধ করুন।



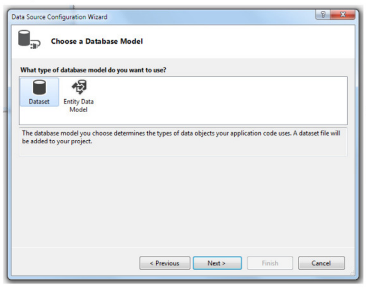
৭. আমাদের ডাটাবেইজ হয়ে গেছে। এবার আমরা ভিজুয়েল এক্সপ্রেস স্টুডিও থেকে একটি নতুন উইন্ডোজ সলুশন তৈরি করবো। এর নামকরণ করুন: 'TikanaDB'। সেইভ করুন।

৮. ফর্মের বায়ের দিকে থাকবে 'Data Sources' নামক একটি ট্যাব। [না থাকলে ক্লিক করুন: **View > Other Windows > Data Sources**]। ক্লিক করুন ডানের চিত্রে দেখানো লিংককে। এবার বায়ের চিত্রে দেখানো জানালাটি আসবে।

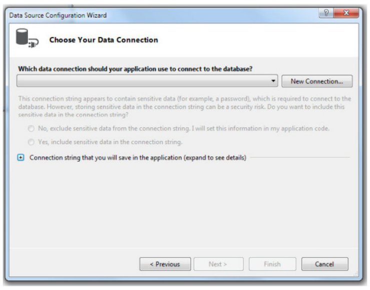




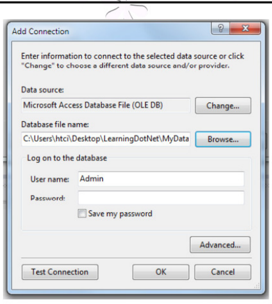
৯. ডাটাবেইজ সিলেক্ট করা থাকবে। Next বোতামে ক্লিক করুন। এর পরে যেসব জানালা আছে সেগুলো একে একে কিভাবে চয়েজ করবেন তা নিচের ক'টি চিত্রে দেখানো হয়েছে। হাইলাইট করা আইটেমগুলো চয়েজ করে করে সর্বশেষ জানালায় যেয়ে 'Finish' বোতামে ক্লিক করুন। ব্যস্ 'TikanaBoi' -এর সঙ্গে কানেকশন হয়ে যাবে।



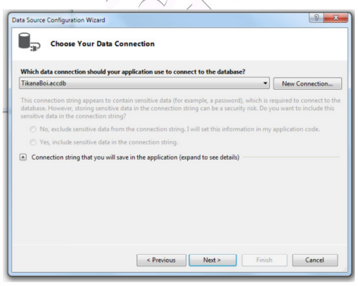
ক্লিক করুন: Next



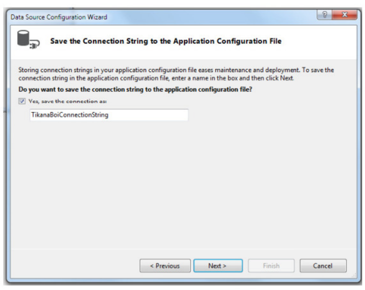
ক্লিক করুন: New Connection...



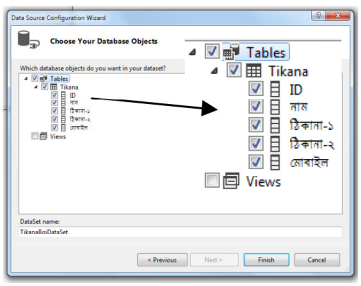
ক্লিক করুন: Browse...



ক্লিক করুন: Next



ক্লিক করুন: Next



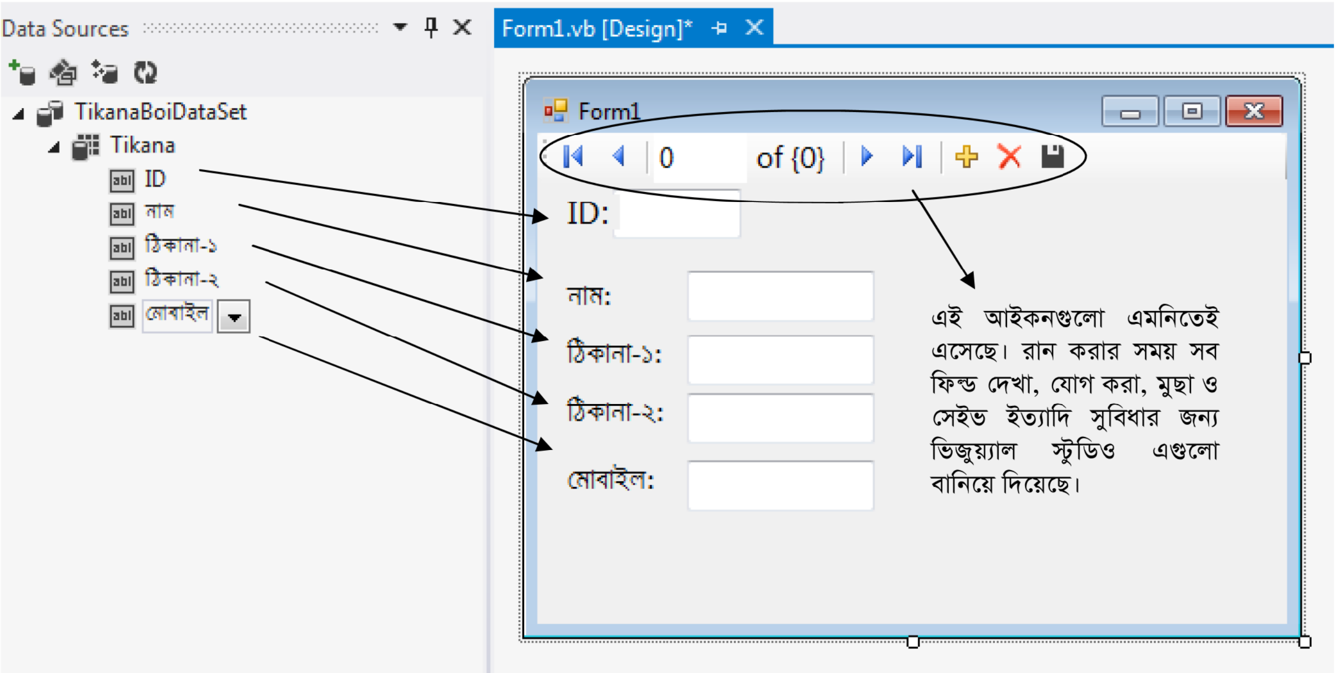
ক্লিক করুন: Finish

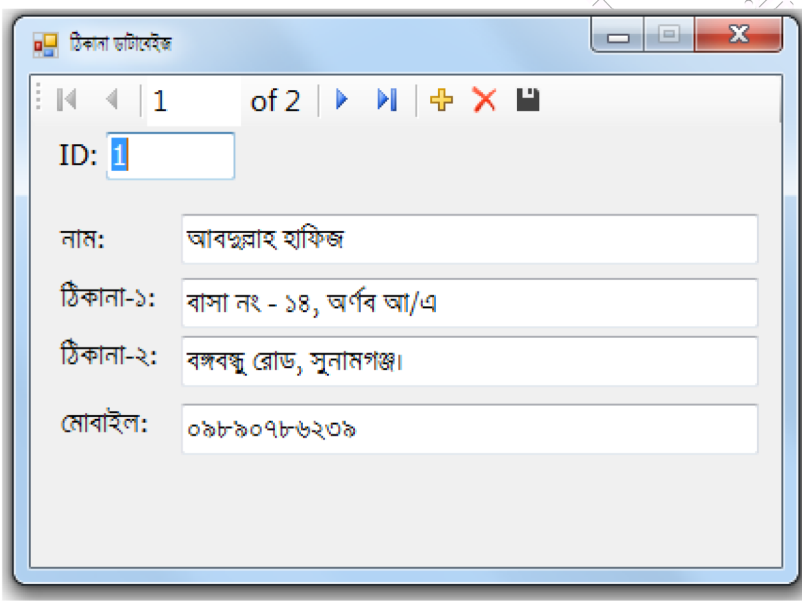
লক্ষ করুন, বায়ের চিত্রে টেবিল ও এর সবগুলো কলাম সিলেক্ট করা আছে। সুতরাং **'Finish'** বোতামে ক্লিকের পূর্বে এগুলো সিলেক্ট করতে ভুলবেন না। আমরা ইনসেটে সিলেকশনকে বড়ো করে দেখিয়েছি। উইজার্ড বন্ধ হওয়ার পর বায়ের ট্যাবের 'Data Sources'-এ ডাটাবেইজের তথ্য আসবে। ***নোট: নিউ কানেকশন [৩নং] ডায়লগ বাদে একটি ডায়লগ আসলে 'No' ক্লিক করবেন।***

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

১০. সফলভাবে কানেকশন হওয়ার পর আমরা ফর্মের মধ্যে টেবিলের বিভিন্ন বস্তু টেনে টেনে নিয়ে এসে যেখানে ইচ্ছে বসাতে পারি। নিচের চিত্র দেখুন। আপনি প্রতিটি আইটেম সিলেক্ট করে লেফ্‌ট-ক্লিক করে টেনে ফর্মের উপর এনে ছেড়ে দেবেন।





লক্ষ করুন, এ পর্যন্ত যাকিছু করলাম, তাতে এক লাইনের কোডিং করতে হয় নি। বাস্তবে ভিজুয়্যাল স্টুডিও আমাদের জন্য কোডিং করে দিয়েছে। কোডিং নিজে না করলেও যাকিছু উইজার্ডের মাধ্যমে হয়েছে- তা-ই আমাদের ডাটাবেইজ দেখার জন্য যথেষ্ট। এ কথার সত্যতা যাচাই করতে হলে এ পর্যায়ে প্রোগ্রামটি একবার রান করুন। বায়ের চিত্রটি দেখুন। আপনি চাইলে + আইকনে ক্লিক করে নতুন রেকর্ড তৈরি করুন। চারটি ফিল্ড পূরণ করে ডানের সেইভ আইকনে ক্লিক দিন। এরপর প্রোগ্রাম বন্ধ করে আবার খুলুন। দেখবেন, তিনটি রেকর্ড দেখাচ্ছে। শেষের [৩নং] রেকর্ডটি দেখুন।

যাক, কোডিং ছাড়াই একটি শক্তিশালী ডাটাবেইজ তৈরি করতে আমরা সক্ষম হয়েছি। তবে আমাদের ইচ্ছে তো, কোডিং শেখা। সুতরাং আসুন, কিভাবে কোডিংয়ের মাধ্যমে অনুরূপ ডাটাবেইজ তৈরি করা যায় তা একবার তলিয়ে দেখি।

১৫তম পরিচ্ছেদ: ***Database Programming Made Simple***

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

ডাটাবেইজ প্রোগ্রাম কোডিং

আগের ক'টি পৃষ্ঠাসমূহে ভিজুয়্যাল স্টুডিও কর্তৃক সৃষ্ট উইজার্ড দ্বারা কিভাবে একটি শক্তিশালী ডাটাবেইজ এ্যাপলিকেশন তৈরি করা যায় তা আমরা দেখেছি। এবার দেখবো ঠিক কিভাবে এ কাজগুলো কোর্ডের মাধ্যমে করা যায়। এ প্রজেক্টের জন্যও আমার Access -এ তৈরি আমাদের 'TikanaBoiDB' ডাটাবেইজ ও এর টেবিলই ব্যবহার করবো। এর একটি কপি আমরা ডেক্সটপে রেখে দেবো। ইউন্ডোজ ফর্ম প্রজেক্ট তৈরিতে নিচের পদক্ষেপগুলো অনুসরণ করুন।

(ক) কোনো প্রজেক্ট খোলা থাকলে বন্ধ করুন। নতুন প্রজেক্ট সৃষ্টি করুন। এর নামকরণ করুন 'TikanaBoiDB2'।

(খ) কানেকশন অবজেক্ট [Connection Object]: যে কোনো ডাটাবেইজের সঙ্গে সংযুক্তির জন্য প্রয়োজন এ বস্তুটি। সুতরাং এ সম্পর্কে আরো জানা জরুরী। এ পরিচ্ছেদের শুরুতেই বলেছি কানেকশন অবজেক্ট কয়েক ধরনের আছে। কোন্ ধরনের কানেকশন অবজেক্ট ব্যবহৃত হবে তা অবশ্যই নির্ভর করে কোন্ ধরনের ডাটাবেইজ নিয়ে আমরা কাজ করবো। আর যেহেতু আমরা Access ডাটাবেইজ নিয়ে কাজ করতে চাই তাই আমাদেরকে ব্যবহার করতে হবে OLE connection object। আগেই বলেছি OLE শব্দের অর্থ হলো, **O**bject **L**inking and **E**mbedding। মূলত এ'টি হচ্ছে অনেকগুলো COM [**C**omponant **O**bject **M**odel] বস্তুর সমষ্টি। সাধারণত অএলই-এর মাধ্যমে ডাটাবেইজের সঙ্গে সংযুক্তি স্থাপিত হয়। তবে এটার মাধ্যমে আরো অনেক সংযুক্তি হয়ে থাকে- যেমন, টেক্সট ফাইল, এসকিউএল সার্ভার, ই-মেইল ইত্যাদি। বেশ ক'টি ভিন্ন OLE অবজেক্ট আছে- এগুলোকে বলে প্রভাইডার [Provider]। আমরা যেটি ব্যবহার করবো তার নাম '**ACE**'।

(গ) যাক, এবার আপনার ফর্মে একটি বোতাম টুলবাক্স থেকে যুক্ত করুন। এর নামকরণ করুন [ইউনিকোডে] ' □□□ □ □□'।

(ঘ) বোতামে ডবল ক্লিক করে এর ক্লিক ইভেন্ট সাবটি খুলুন। এতে যুক্ত করুন নিচের কোড লাইনটি:

```
Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
```

(ঙ) কানেকশন অবজেক্টের সঙ্গে সম্পৃক্ত আছে বিভিন্ন প্রোপার্টি। এর মধ্যে একটি হচ্ছে ConnectionString প্রেপার্টি। এটা অনেক প্যারামিটার নিতে পারে। আমাদের ক্ষেত্রে অবশ্য কয়েকটি মাত্র দরকার। যেমন: প্রথমত একে বলতে হবে ডাটাবেইজ কানেকশন হবে ও দ্বিতীয়ত কোথায় সে ডাটাবেইজ আছে তা অবগত করা। ভিজুয়্যাল বেসিক.নেট প্রথমটির জন্য যা দিয়েছে তার নাম Provider এবং দ্বিতীয়টির জন্য দিয়েছে Data Source। সুতরাং, এবার আমরা উপরের লাইনের পরে নিচের ৪টি লাইন লিখবো বোতামের ক্লিক ইভেন্টে। এর প্রথম তিনটি স্ট্রিং ভেরিয়েবল ডিক্লারেশন। বাকি একটি ভেরিয়েবল এসাইনমেন্ট।

```
Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে [দেখুন ৬ষ্ঠ লাইনটি]।
Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
```

লক্ষ করুন, উপরের শেষের লাইনের কোড হতে পারে: dbProvider = "PROVIDER=Microsoft.**ACE.OLEDB.15.0**;" -কোন্‌টি হবে তা নির্ভর করবে আমাদের ভার্সনের ওপর। আমাদের ব্যবহৃত ভার্সনে যদি কাজ হয় না- আপনি দ্বিতীয়টি টেস্ট করে দেখতে পারেন। এবার নিচের লাইনটি লিখুন:

```
FullDatabasePath = "C:\Users\htci\Desktop\TikanaBoi.accdb" 'ডাটাবেইজের পুরো প্যাথ নাম।
```

এবার নিচের লাইনটি লিখুন:

```
dbSource = "Data Source=" & FullDatabasePath 'dbSource মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
```

এ পর্যন্ত কোডিংয়ে কি লিখা হলো তা নিচে তুলে ধরছি। লক্ষ করুন ডিবাগের জন্য ব্যবহৃত মেসেজবক্স ফাংশনটি। এটা অবশ্য পরে কমেন্ট দ্বারা বা মুছে অকেজো করা হবে।

```
Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে [দেখুন ৬ষ্ঠ লাইনটি]।
Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
```

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
FullDatabasePath = "C:\Users\htci\Desktop\TikanaBoi.accdb" 'ডেক্সটপে থাকা ডটাবেইজের পুরো প্যাথ নাম।
dbSource = "Data Source=" & FullDatabasePath 'dbSource মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
'MsgBox(dbSource) 'ডিবাগ কারর জন্য এই লাইন। এটি থাকবে না।
```

এ পর্যন্ত কোড দ্বারা আমরা dbSource স্ট্রিং ভেরিয়েবলে কি হবে তা নিশ্চিত করেছি। অর্থাৎ ডাটা সূত্রের প্যাথ নাম সংরক্ষণ করেছি। কোডের প্রথম লাইনে আমরা যে অবজেক্ট ভেরিয়েবল ডিক্লায়ার করেছিলাম এখন তাকে কাজে লাগাতে হবে। এজন্য একে দু'টি তথ্য দেওয়া প্রয়োজন: ১. ডাটাবেইজ প্রভাইডার ও ২. ডাটাসূত্র। আমরা উভয়টিই নির্দিষ্ট করে নিয়েছি। অর্থাৎ ডাটাবেইজ প্রভাইডার হলো: `"PROVIDER=Microsoft.ACE.OLEDB.12.0;"` [dbProvider] এবং ডাটাসূত্র হলো `"Data Source = " & FullDatabasePath` [dbSource]। আর এ উভয় স্ট্রিং মিলেই তৈরি হয় কানেকশন স্ট্রিংটি। সুতরাং আমরা এবার এ লাইনটি কোডে লিখতে পারি:

con.ConnectionString = dbProvider & dbSource

অবশেষে আমাদের কানেকশন স্ট্রিং প্রস্তুত। এখন অতি সহজেই আমাদের ডাটাবেইজ খুলতে পারি। এজন্য একটি মাত্র ছোট্ট কোড লাইনের প্রয়োজন:

con.Open( ) -এ কোডটুকু হলো `OleDb.OleDbConnection` অবজেক্টের একটি ফাংশন।

ডাটাবেইজ খোলার পর যাকিছু ইচ্ছে তা সেরে একে আবার ক্লোজ করতে হবে। এটা খুব গুরুত্বপূর্ণ ব্যাপার। ক্লোজ করার জন্যও আমরা আরেকটি সহজ ফাংশন ব্যবহার করে থাকি:

con.Close( ) -ব্যস! ডাটাবেইজ খোলা ও বন্ধ করার কোড শেষ হলো।

এ পর্যন্ত কোডে যা হয়েছে তাহলো:

```
Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে [দেখুন ৬ষ্ঠ লাইনটি]।
Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
FullDatabasePath = "C:\Users\htci\Desktop\TikanaBoi.accdb" 'ডেক্সটপে থাকা ডটাবেইজের পুরো প্যাথ নাম।
dbSource = "Data Source=" & FullDatabasePath 'dbSource মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
'MsgBox(dbSource) 'ডিবাগ কারর জন্য এই লাইন। এটি থাকবে না।
con.Open( ) 'ডাটাবেইজ খোলা।
con.Close( ) 'ডাটাবেইজ বন্ধ করা।
```

আমরা কাঙ্ক্ষিত গন্তব্যের অর্ধেক পথে এসে গেছি। এবার আরো অগ্রসর হতে হবে- অর্থাৎ, ডাটাবেইজ প্রদর্শনের দরকার এবং বিভিন্ন তথ্য দেখা ও প্রয়োজনে আপডেট করা। পরের পৃষ্ঠাসমূহে আমরা এ সম্পর্কিত কোডগুলো লিখবো। শিক্ষার্থীরা ধৈর্যসহ আমার সাথেই থাকুন এবং স্টাডি করুন।

ডাটাসেট [DataSet] ও ডাটা এডাপটার [Data Adapter]

ডাটা বা তথ্য ডাটাবেজ থেকে কম্পিউটারের মেমোরিতে সংরক্ষণ করার পদ্ধতি হিসাবে আমরা যে সিস্টেম ফাংশন ব্যবহার করি তাকে ADO.Net নামকরণ করেছে DataSet । এটা টেবিলের তথ্যাদির একটি কপি মেমোরিতে হোল্ড করে রাখে। এটা এমন কিছু নয় যা আমাদের ফর্মে লিখে দেখানো যাবে। তবে মেমোরির মধ্যেও এটি কাল্পনিক টেবিল হিসাবে থাকে- অর্থাৎ কলাম ও রো-তে সব তথ্য সংরক্ষিত হয়। একে ডাটাবেইজের ভাষায় ফিল্ড সিস্টেম বলে। তবে প্রশ্ন হচ্ছে DataSet ও Connection Object কি একে অন্যকে দেখতে পারে- বা এমনকি যোগাযোগ করতে সক্ষম?

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

এর জবাব হলো, না। এরা আলাদা। সুতরাং এদের মধ্যকার কমিউনিকেশন ফাঁকটি কোনো উপায়ে পূরণ করতে হবে। আর এ কাজের জন্যই তৈরি হয়েছে Data Adapter।

Data Adapter হলো DataSet ও Connection Object-এর মধ্যকার যোগাযোগ সৃষ্টিকারী মাধ্যম। ডাটা এডাপটার আপনার কানেকশন অবজেক্টের সঙ্গে যোগাযোগ করে এবং যে অনুসন্ধান আপনি সেট করেন তা কার্যকর করে দেয়। ফলাফল ডাটা সেটে সংরক্ষিত হয়। ডাটা সেট ও ডাটা এডাপটার উভয়টিই অবজেক্ট টাইপ। সুতরাং এদেরকে সেট করতে আমরা ব্যবহার করি নিচের ডিক্লারেশগুলোর মতো:

```
Dim ds As New DataSet
Dim da As OleDb.OleDbDataAdapter
```

এরপর:

```
da = New OleDb.OleDbDataAdapter(sql, con)
```

এ কোডগুলোর ব্যাখ্যা এরূপ: DataSet একটি অবজেক্ট। তাই একে New কিওয়ার্ড দ্বারা ডিক্লারেশন করেছি। এই অবজেক্ট বা বস্তুটি সিস্টেম অবজেক্ট। আর সিস্টেম অবজেক্ট ব্যবহার করতে হলে প্রথমে অবজেক্টের একটি ইন্সট্যান্ট [বা কপি] New কিওয়ার্ড দ্বারা ডিক্লায়ার করতে হয়। অপরদিকে DataAdapter হচ্ছে OLEDB অবজেক্টের একটি প্রোপার্টি বা বৈশিষ্ট্য। সুতরাং আমাদেরকে . [ডট] ব্যবহার করতে হয়েছে ইঙ্গিত [বা পয়েন্ট] করার উদ্দেশ্যে। অন্যকথায় da নামক ভেরিয়েবলে এই অবজেক্টটি পাস [pass] করছি। এই ভেরিয়েবল এখন DataAdapter -এর একটি রেফারেন্স হোল্ড করবে।

যাক, দ্বিতীয় লাইন DataAdapter এর রেফারেন্স হোল্ড করবে আর তৃতীয়টি সৃষ্টি করবে একটি নতুন DataAdapter অবজেক্ট। আমরা এর ব্র্যাকেট দু'টোর মধ্যে দু'টি কোড লিখেছি: sql, con। প্রথমটি হলো Structured Query Language- এর একটি কমান্ডের জন্য সৃষ্ট ভেরিয়েবল এবং দ্বিতীয়টি হলো আমাদের কানেকশন অবজেক্ট। এই অবজেক্টটি আমরা ইতোমধ্যে ডিক্লায়ার করেছি। এবার আমাদের সৃষ্ট ভেরিয়েবল sql-কে সেট করতে হবে। একটু পরই সেখানে আসছি। প্রথমে আবারো উল্লেখ করা জরুরী যে: DataAdapter এর কাজ হলো- কানেকশন অবজেক্ট ও ডাটাসেটের মধ্যে কমিউনিকেশন এজেন্ট হিসাবে কাজ করা।

ডাটাবেইজ প্রোগ্রামিংয়ের ভাষা: Structured Query Language [SQL]

ইতোমধ্যে আমরা বেশ কয়েকবার sql [উচ্চারণ, সিকুয়েল] শব্দটি নিয়ে কথা বলেছি। তবে এর ব্যাখ্যা দিই নি। বাস্তবে এটা একটি ডাটাবেইজ প্রোগ্রামিং ভাষার নাম। এ ভাষা সম্পর্কে অন্তত কিছুটা জ্ঞান না থাকলে সত্যিকার অর্থে কোডের মাধ্যমে যে কোন ডাটাবেইজ নিয়ে কাজ করা আদৌ সম্ভব নয়। আমরা এখানে Access ডাটাবেইজ নিয়ে কাজ করছি। সিকুয়েল দ্বারা এরূপ ডাটাবেইজকে নিয়ন্ত্রণ করা যায়। তবে আরো কিছু ডাটাবেইজ আছে যেগুলোর সঙ্গে যোগাযোগ সৃষ্টিতে এই একই ভাষা ব্যবহার করা সম্ভব। এ গ্রন্থে আমরা সেদিকে যাচ্ছি না। শুধু Access ডাটাবেইজের ওপরই আমাদের আলোচনা সীমিত রাখবো। ভাষার মৌলিক নির্দেশনা বেশ সহজ। যেমন:

SELECT * FROM TableName -এই লাইনের অর্থ ডাটাবেইজের টেবিলের পুরো রেকর্ড। লক্ষ করুন **TableName** সিকুয়েলের নির্দেশনা নয়- এটা হচ্ছে যে টেবিল নিয়ে কাজ করবেন তার নাম। যেমন, আমাদের ক্ষেত্রে এটি হবে: Tikana। * অর্থ 'All Records'। সিকুয়েলের সকল কিওয়ার্ড বড়ো অক্ষরে লেখা উত্তম। ফলে আমরা বুঝতে পারবো কোন্‌টি কিওয়ার্ড ও কোন্‌টি নয়। সুতরাং এ লাইনটির অর্থ হলো: **TableName** নামক টেবিল থেকে সকল রেকর্ড সিলেক্ট করো। তবে * -এটা দিয়ে টেবিলের সকল রেকর্ড সিলেক্ট করা ছাড়াও আমরা চাইলে শুধুমাত্র একটি কলাম বা রো সিলেক্ট করতে পারি। যেমন:

```
SELECT Tikana.নাম, Tikana.মোবাইল FROM Tikana
```

আমাদের উদ্দেশ্য কিন্তু পুরো টেবিলটিই সিলেক্ট করা। সুতরাং সিকুয়েলের অন্যান্য কমান্ড সম্পর্কে আর অতিরিক্ত কিছু বলছি না। শিক্ষার্থীরা চাইলে সিকুয়েল বিষয়ের ওপর গ্রন্থাদি পাঠ করতে পারেন। ইন্টারনেটেও এ ভাষা শেখার জন্য

১৫তম পরিচ্ছেদ: ***Database Programming Made Simple***

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

অনেক ওয়েভপেইজ আছে। যাক, আমরা এখন তিনটি ডিক্লারেশন ও দু'টি লাইন বোতামের কোডে যুক্ত করতে চাই:

```
'=======নতুন ডিক্লারেশন ==========
    Dim ds As New DataSet 'একটি ডাটাসেট অবজেক্ট হোল্ড করে।
    Dim da As OleDb.OleDbDataAdapter 'একটি এডাপটার অবজেক্ট হোল্ড করে।
    Dim Sql As String 'একটি সিকুয়েল স্ট্রিং হোল্ড করে।
'==========================
'======= নতুন কোড ========
    Sql = "SELECT * FROM Tikana" 'সিকুয়েল স্ট্রিং স্টোর করা।
    ' ডাটা এডাপটারের নিকট সিকুয়েল স্ট্রিং ও কানেকশন অবজেক্ট পাস করে দেওয়া।
    da = New OleDb.OleDbDataAdapter(Sql, con)
'=====================
```

এখন পর্যন্ত বোতামের কোড হলো:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
    Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে [দেখুন ১০ম লাইনটি]।
    Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
    Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
    '=======নতুন ডিক্লারেশন ==========
    Dim ds As New DataSet 'একটি ডাটাসেট অবজেক্ট হোল্ড করে।
    Dim da As OleDb.OleDbDataAdapter 'একটি এডাপটার অবজেক্ট হোল্ড করে।
    Dim Sql As String 'একটি সিকুয়েল স্ট্রিং হোল্ড করে।
    '==========================
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
    ' MsgBox(CurDir)
    FullDatabasePath = "C:\Users\htci\Desktop\TikanaBoi.accdb" 'ডাটাবেইজের পুরো প্যাথ নাম।
    dbSource = "Data Source=" & FullDatabasePath 'dbSource মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
    'MsgBox(dbSource) 'ডিবাগ কারর জন্য এই লাইন। এটি থাকবে না।
    con.ConnectionString = dbProvider & dbSource
    con.Open()
    '======= নতুন কোড ========
    Sql = "SELECT * FROM Tikana" 'সিকুয়েল স্ট্রিং স্টোর করা।
    ' ডাটা এডাপটারের নিকট সিকুয়েল স্ট্রিং ও কানেকশন অবজেক্ট পাস করে দেওয়া।
    da = New OleDb.OleDbDataAdapter(Sql, con)
    '=====================
    con.Close()
End Sub
```

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

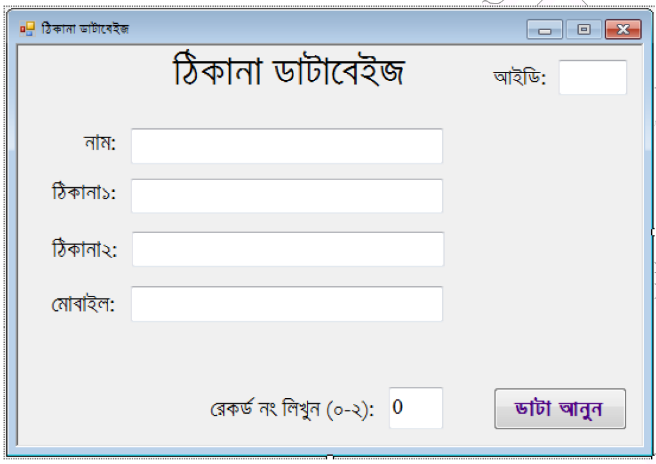
সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

এবার ডাটাসেট পূরণ করার পালা। এজন্য প্রয়োজন আমাদের ডিক্লায়ার করা ডাটা এডাপটার- তথা da। সুতরাং কোড ব্লকের লাইন: da = New OleDb.OleDbDataAdapter(Sql, con) -এর পরই আমাদেরকে নিচের লাইনটি লিখতে হবে।

da.Fill(ds, "TikanaDB") '[illegible]।

আমাদের ডাটা এডাপটার সৃষ্টি হয়ে গেছে। এখন DataSet(ds) -টি আমাদের বানানো TikanaBoi.accdb ডাটাবেইজের **Tikana** নামক টেবিলটি পুরোপুরি হোল্ড করবে। তবে এক সমস্যা তো রয়েই গেলো! কেউ তো তথ্যাদি দেখতে পারবেন না। এ ব্যাপারে আরো কিছু কাজ করতে হবে। আর তা আমরা এক্ষুণি করবো।

টেবিলের তথ্যাদি প্রদর্শনের জন্য প্রয়োজন কয়েকটি টেক্সট্ বাক্স। সুতরাং নিচের চিত্র অনুযায়ী সবক'টি টেক্সট্ বাক্স ও লেবেল ফর্মে নিয়ে যুক্ত করুন।



লক্ষ করুন: চারটি খালি টেক্সট্ বাক্সে থাকবে টেবিলের তথ্যাদি। উপরের ছোট্ট টেক্সট্ বাক্সে থকাবে আইডি নাম্বার। নিচের ছোট্ট টেক্সট্ বাক্সে থাকবে রেকর্ড নাম্বার। যেহেতু রেকর্ড শুরু হয় ০ থেকে তাই এটাই থাকবে ডিফোল্ট হিসাবে। আমাদের কোড এই সংখ্যাটি নিয়ে তথ্য প্রদর্শন করবে। ব্যবহারকারী যখন ১ কিংবা ২ লিখবেন এবং বোতামে টিপ দিবেন তখন এই সংখ্যা মুতাবিক রেকর্ড প্রদর্শন হবে। যাক যে অতিরিক্ত লাইনগুলো বোতামের ক্লিক ইভেন্টে লিখা হবে সেগুলো নিচে তুলে ধরছি। এরপর পুরো কোডটি আমরা তুলে ধরবো।
যেহেতু আমাদের টেবিলে প্রত্যেক রেকর্ডে ৪টি করে রেকর্ড আছে তাই এগুলো প্রদর্শনের জন্য চারটি ফিল্ডের কথাই লিখতে হবে। তবে সর্বপ্রথম আমাদেরকে জানতে হবে, ব্যবহারকারী যে সংখ্যাটি রেকর্ড নং ফিল্ডে লিখেছেন তা সঠিক কি না।

```
'validate record number
If Val(TextBox1.Text) < 0 Or Val(TextBox1.Text) > 2 Then 'ভেলিড সংখ্যা না হলে
    MsgBox("Please enter 0, 1 or 2 only") 'এই মেসেজ প্রদর্শন করে সাব থেকে বেরিয়ে যাওয়া।
    Exit Sub
End If
'===============
    Dim RecNum As Integer = 0 'এটা হোল্ড করবে টেক্সট্ বাক্সের সংখ্যাটি যার দ্বারা সঠিক রো এর তথ্য আনা হবে।
    RecNum = Val(TextBox1.Text) 'ভেরিয়েবলে টেক্সবাক্সের সংখ্যা [০,১ বা ২]
```

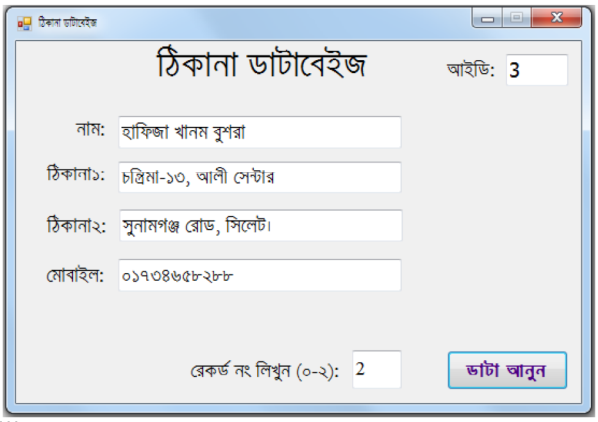
উক্ত কোড ব্লক কিভাবে কাজ করে তা আশারাখি বুঝিয়ে বলার প্রয়োজন নেই। If -- End if ব্লকের কাজ হলো টেক্সট্ বাক্স-১ এর ভেতর ব্যবহারকারী সঠিক সংখ্যা [০,১ বা ২] লিখেছেন কি না। টেস্ট ফেইল হলে একটি মেসেজ বাক্স দেখিয়ে বোতামের ইভেন্ট সাব থেকে বেরিয়ে আসার ব্যবস্থা করা হয়েছে। এরপর আবার সঠিক সংখ্যা লিখলে কিন্তু টেবিল থেকে চারটি ভ্যালু উপরের চারটি টেক্সট্ বাক্সে প্রদর্শিত হবে। লক্ষ করুন, ডিফোল্ট সংখ্যা হিসাবে ০ রেখেছি। টেক্সবাক্স-১ এর Text প্রোপার্টিতে ডিজাইন-টাইমে ০ লিখে দিয়েছি। অপর আরেক ব্যাপার হলো উপরের চারটি টেক্সবাক্সকে রীড-অনলি বাক্সে পরিণত করেছি যাতেকরে কেউ কিছু এতে লিখতে পারেন না। প্রোপার্টিতে যেয়ে টেক্সবাক্সের ReadOnly প্রোপার্টি True করলেই কাজ হয়ে যাবে। যাক এবার আমরা পরের পৃষ্ঠায় পূর্ণ কোডটি তুলে ধরছি। শিক্ষার্থীরা কোডগুলো দেখে দেখে ভালো করে বুঝার চেষ্টা করবেন। অবশ্য কোড লেখার পর আমরা প্রোগ্রামটি রানকালীন একটি জানালার চিত্রও দেখিয়েছি। আমাদের ডাটাবেইজের আইটেমগুলো সঠিকভাবে আসছে কি না তা তো টেস্ট করে দেখা চাই। ঠিক না?

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
Public Class Form1
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      'validate record number
      If Val(TextBox1.Text) < 0 Or Val(TextBox1.Text) > 2 Then 'ভেলিড সংখ্যা না হলে
          MsgBox("Please enter 0, 1 or 2 only") 'এই মেসেজ প্রদর্শন করে সাব থেকে বেরিয়ে যাওয়া।
     Exit Sub
       End If
       '==============
        Dim RecNum As Integer = 0 'এটা হোল্ড করবে টেক্সট বাক্সের সংখ্যাটি যার দ্বারা সঠিক রো এর তথ্য আনা হবে।
        RecNum = Val(TextBox1.Text) 'ভেরিয়েবলে টেক্সবাক্সের সয়খ্যা [০,১ বা ২]
        Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
        Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে।
        Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
        Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
        Dim ds As New DataSet 'একটি ডাটাসেট অবজেক্ট হোল্ড করে।
        Dim da As OleDb.OleDbDataAdapter 'একটি এডাপটার অবজেক্ট হোল্ড করে।
        Dim Sql As String 'একটি সিকুয়েল স্ট্রিং হোল্ড করে।
       '===========================
        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
        FullDatabasePath = "C:\Users\htci\Desktop\TikanaBoi.accdb" 'ডাটাবেইজের পুরো প্যাথ নাম।
        dbSource = "Data Source=" & FullDatabasePath 'dbSource এর মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
        'MsgBox(dbSource) 'ডিবাগ কারর জন্য এই লাইন। এটি থাকবে না।
        con.ConnectionString = dbProvider & dbSource
        con.Open() 'ডাটাবেইজটি খোলার নিরর্দশ।
        Sql = "SELECT * FROM Tikana" 'সিকুয়েল স্ট্রিং স্টোর করা।
        'ডাটা এডাপটারের নিকট সিকুয়েল স্ট্রিং ও কানেকশন অবজেক্ট পাস করে দেওয়া।
        da = New OleDb.OleDbDataAdapter(Sql, con)
        da.Fill(ds, "TikanaDB") 'দ্বিতীয় প্যারামিটার যে কোনো কিছু হতে পারে।
        IDtext.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(0) 'আইডি
        NameText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(1) 'নাম
        AddressText1.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(2) 'ঠিকান-১
        AddressText2.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(3) 'ঠিকানা-২
        MobileText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(4) 'মোবাইল
       '======================
        con.Close() 'ডাটাবেইজটি বন্ধের নির্দেশ।
   End Sub
End Class
```

ডানের চিত্রে ঠিকানা ডাটাবেইজ প্রোগ্রামটি রানকালীন জানালার একটি দৃশ্য দেখাচ্ছে। লক্ষ করুন, রেকর্ড দেখার জন্য আমাদেরকে একটি ইন্টেজার [০,১,২] প্রবেশ করতে হবে। কিন্তু আমরা যখন উইজার্ড দিয়ে একই ডাটাবেইজ তৈরি করেছিলাম, তখন ন্যাভিগেট করার জন্য দু'টি বোতাম ছিলো। ওগুলোতে ক্লিক করে করে আমরা এক রেকর্ড থেকে অন্যটিতে যাই। আর এটাই স্বাভাবিক। সুতরাং কোড দ্বারা সৃষ্ট আমাদের ঠিকানা ডাটাবেইজও কি অনুরূপ প্রাকটিক্যাল বানাতে পরি? হ্যাঁ, পারি। সুতরাং আসুন, এবার ন্যাভিগেশন কোডগুলো কিভাবে লিখতে হবে তা জেনে নিই। প্রথমে ফর্মের ডিজাইনে কিছু পরিবর্তন আনতে হবে। নিচের চিত্রে সেটা তুলে ধরেছি।

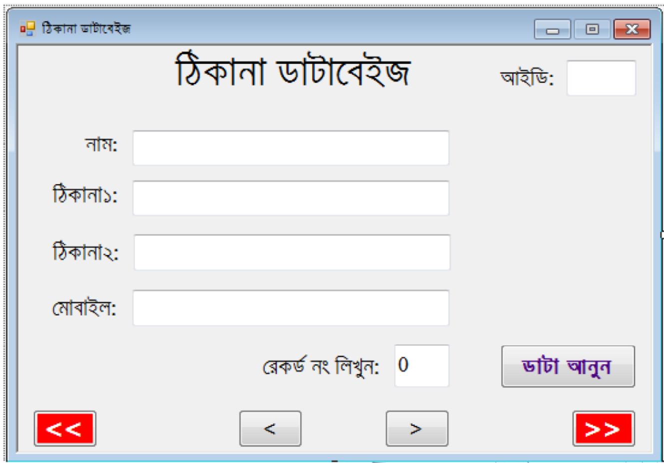


এখানে আমরা যা করেছি তাহলো, চারটি ন্যাভিগেশন বোতাম যুক্ত করেছি। সর্বাধিক ডানের ও বামের বোতাম দু'টো হলো যথাক্রমে

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

শেষের রেকর্ড ও প্রথম রেকর্ড দেখার বোতাম। মাঝের দু'টো প্রদর্শিত রেকর্ড থেকে পরেরটা ও আগেরটায় যাওয়ার বোতাম। কোডের সুবিধার্তে প্রতিটি বোতামের নামকরণ করেছি এভাবে:



সর্বাধিক ডানের বোতাম: btnLast।
সর্বাধিক বামের বোতাম: btnFirst।
পরের রেকর্ডের বোতাম: btnNext।
আগের রেকর্ডের বোতাম: btnPrev।

আমাদের ইচ্ছে ঠিকানা ডাটাবেইজটি ব্যবহারবান্ধব হবে। সুতরাং আমরা চাই রেকর্ড দেখার জন্য রেকর্ড সংখ্যা দ্বারা যেরূপ রেকর্ডের তথ্য নিয়ে আসা যায় সেরূপ যাতে ন্যাভিগেশন বোতাম দ্বারাও রেকর্ড দেখার সুযোগ থাকে। এছাড়া প্রথম ও শেষ রেকর্ড দেখারও সহজ উপায় থাকতে হবে। এ কারণেই 'ডাটা আনুন' বোতাম এবং সংখ্যা প্রবেশের ফিল্ডটি রেখে দিয়েছি।

ন্যাভিগেশনের জন্য নতুন কোড লিখতে হবে। তবে আমরা ইতোমধ্যে যাকিছু কোড করেছি তার বেশিরভাগই ফাইনাল কোডে থাকবে। নিচে এবার পুরো কোডটি তুলে ধরেছি। এরপর অতিরিক্ত কোড কিভাবে কাজ করে তা একে একে বুঝিয়ে দেবো। এই ব্যাখ্যা শেষে Database Programming এর মতো বিরাট বিষয়ের ওপর আমাদের প্রাথমিক অনুসন্ধানমূলক এ পরিচ্ছেদের ইতি ঘটবে। শিক্ষার্থীরা এ থেকেই যা শিখেছেন, সাধারণ কাজের জন্য তা-ই যথেষ্ট হবে। ইন্টারনেটে অনেক স্যাম্পুল ডাটাবেইজ এ্যাপলিকেশন পাওয়া যায়। যারা এ বিষয়ে আরো গভীরে যেতে ইচ্ছুক তাদের প্রতি আমার উপদেশ হলো ইন্টারনেটের এসব স্যাম্পুল ডাউনলোড করে কোড দেখুন ও শিখুন।

ঠিকানা ডাটাবেইজের এ পর্যন্ত পূর্ণাঙ্গ কোড

```
Public Class Form1
    'গ্লোবাল ডিক্লারেশন
    Dim RecNum As Integer = 0 'এটা হোল্ড করবে টেক্সট বাক্সের সংখ্যাটি যার দ্বারা সঠিক রো এর তথ্য আনা হবে।
    Dim NumRec As Integer 'এটা হোল্ড করবে টেবিলের মোট রেকর্ড নাম্বার।
    'এটা হলো ডাটা আনুন বোতামের সাবরুটিন। এটা কল করে GetAndFillRecord সাব।
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        GetAndFillRecord() 'সাব কল।
    End Sub
    'এই সাব ন্যাভিগেশন বোতামগুলো সক্রিয় করে। প্রাথমিকভাবে ওগুলো অক্রিয়।
    Private Sub EnableNavigationBtn()
        btnFirst.Enabled = True : btnLast.Enabled = True
        btnNext.Enabled = True : btnPrev.Enabled = True
   End Sub
    'এ ফাংশনের কাজ হলো টেক্সট বাক্সে সঠিক সংখ্যা আছে কি না, তা দেখা। রিটার্ন করবে বুলিয়ান ভ্যালু [সত্য বা মিথ্যা]।
    Private Function ValidateEntryNum()
        If Val(TextBox1.Text) < 0 Or Val(TextBox1.Text) > NumRec - 1 Then 'ভেলিড সংখ্যা না হলে
            'এই ফাংশন কলের পূর্বে অন্তত একবার ডাটাবেইজ খোলা হবে। ফলে NumRec এর মান কি তা-ও জানা থাকবে
        MsgBox("There are 0 to " & NumRec - 1 & " records only!") 'এই মেসেজ প্রদর্শন করে ফলস ভ্যালু রিটার্ন করা।
            Return False
        End If
        Return True 'গ্রহণযোগ্য সংখ্যা- তাই রিটার্ন ট্রু।
    End Function
    'রেকর্ড থেকে তথ্য এনে ফিল্ডে দেখানো হলো এ সাবের কাজ
    Private Sub GetAndFillRecord()
```

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
        RecNum = Val(TextBox1.Text) 'ভেরিয়েবলে টেক্সবাক্সের সংখ্যা। প্রথমবার এর মান ০।
        Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
        Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে।
        Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
        Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম।
        Dim ds As New DataSet 'একটি ডাটাসেট অবজেক্ট হোল্ড করে।
        Dim da As OleDb.OleDbDataAdapter 'একটি এডাপটার অবজেক্ট হোল্ড করে।
        Dim Sql As String 'একটি সিকুয়েল স্ট্রিং হোল্ড করে।
        dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
        FullDatabasePath = "C:\Users\htci\Desktop\LearningDotNet\MyDatabase\TikanaBoi.accdb" 'ডাটাবেইজের পুরো প্যাথ নাম।
        dbSource = "Data Source=" & FullDatabasePath 'dbSource এর মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
        con.ConnectionString = dbProvider & dbSource 'কানেকশন স্ট্রিং পূর্ণ হলো।
        con.Open() 'ডাটাবেইজ খোলার নির্দেশ।
        'ক টি রেকর্ড আছে তা জানা। ভেরিয়েবল cmd একটি সিকুয়েল কমান্ড হোল্ড করবে।
        Dim cmd As New OleDb.OleDbCommand("Select COUNT(*) FROM Tikana", con)
        NumRec = cmd.ExecuteScalar() 'সিস্টেম ফাংশন দ্বারা রেকর্ড সংখ্যা আনা।
        'ফাংশন কল করে রেকর্ড সংখ্যা সঠিক আছে কি না দেখা।
        If ValidateEntryNum() = False Then
            con.Close() 'ভুল হলে ডাটাবেইজ বন্ধ করে এক্সিট করা
            Exit Sub
        End If
        Sql = "SELECT * FROM Tikana" 'সিকুয়েল স্ট্রিং স্টোর করা।
        'ডাটা এডাপটারের নিকট সিকুয়েল স্ট্রিং ও কানেকশন অবজেক্ট পাস করে দেওয়া।
        da = New OleDb.OleDbDataAdapter(Sql, con)
        da.Fill(ds, "TikanaDB") 'দ্বিতীয় প্যারামিটার যে কোনো কিছু হতে পারে।
        IDText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(0) 'আইডি
        NameText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(1) 'নাম
        AddressText1.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(2) 'ঠিকান-১
        AddressText2.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(3) 'ঠিকানা-২
        MobileText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(4) 'মোবাইল
        EnableNavigationBtn() 'ডাটাবেইজ লোড হয়েছে তাই চারটি ন্যাভিগেশন বোতাম সক্রিয় করতে হবে।
        '======================
        con.Close() 'ডাটাবেইজ ক্লোজ করা- এটা জরুরী।
    End Sub
    'এটা প্রথম রেকর্ড দেখার বোতামের সাবরুটিরন।
    Private Sub btnFirst_Click(sender As Object, e As EventArgs) Handles btnFirst.Click
        TextBox1.Text = "0" 'ট্যাক্সট বাক্সে প্রথম রেকর্ড ইনডেক্স সংখ্যা লিখে দেওয়া।
        GetAndFillRecord() 'উপলের রেকর্ড প্রদর্শনের সাবরুটিন কল করা।
    End Sub
    'এটা সর্বশেষ রেকর্ড দেখার বেতামের সাবরুটিন।
    Private Sub btnLast_Click(sender As Object, e As EventArgs) Handles btnLast.Click
        TextBox1.Text = NumRec - 1 'ইনডেক্স শুরু হয় ০ থেকে তাই মোট রেকর্ড থেকে ১ কমাতে হবে।
        GetAndFillRecord() 'রেকর্ড প্রদর্শনের সাবরুটিন কল করা।
    End Sub
    'এটা পরের রেকর্ড দেখার বোতামের সাবরুটিন।
    Private Sub btnNext_Click(sender As Object, e As EventArgs) Handles btnNext.Click
        If ValidateEntryNum() <> False Then 'প্রথমে ভেলিড সংখ্যা আছে কি না দেখতে হবে।
            If TextBox1.Text >= NumRec Then 'ভেলিড সংখ্যা সত্য হলেও, সর্বশেষ রেকর্ডে আসা হয়েছে কিনা দেখতে হবে।
                Beep() 'সর্বশেষ রেকর্ড এটি- সিস্টেম সাউন্ড প্লে করে সাব থেকে বেরিয়ে যাওয়া।
                Exit Sub
            End If
```

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
        Dim entryNum As Integer = Val(TextBox1.Text) + 1 'টেক্স বাক্সের সংখ্যা ১ বাড়ানো।
            If entryNum >= NumRec Then 'আবার চেক করা দরকার, এ নতুন সংখ্যা রেকর্ড সংখ্যা থেকে বেশি কি না।
                Beep() 'বেশি হলে বিপ বাজিয়ে আউট!
                Exit Sub
            End If
            TextBox1.Text = entryNum 'নতুন নাম্বার টেক্স বাক্সে নেওয়া ও...
            GetAndFillRecord() 'প্রদর্শন সাব কল করা।
        End If
    End Sub
    'এটা আগের রেকর্ড দেখার বোতামের সাবরুটিন।
    Private Sub btnPrev_Click(sender As Object, e As EventArgs) Handles btnPrev.Click
        If ValidateEntryNum() <> False Then 'প্রথমে ভেলিড সংখ্যা আছে কি না দেখতে হবে।
            If TextBox1.Text <= 0 Then 'ভেলিড সংখ্যা সত্য হলেও, প্রথম রেকর্ডে আসা হয়েছে কিনা দেখতে হবে।
            Beep() 'প্রথম রেকর্ড এটি- সিস্টেম সাউন্ড প্লে করে সাব থেকে বেরিয়ে যাওয়া।
            Exit Sub
        End If
        Dim entryNum As Integer = Val(TextBox1.Text) - 1 'টেক্স বাক্সের সংখ্যা ১ কামনো।
        If entryNum < 0 Then 'আবার চেক করা দরকার, এ নতুন সংখ্যা শূন্য বা প্রথম রেকর্ড সংখ্যা থেকে কম কি না।
            Beep() 'কম হলে বিপ বাজিয়ে আউট!
            Exit Sub
        End If
        TextBox1.Text = entryNum 'নতুন নাম্বার টেক্স বাক্সে নেওয়া ও...
        GetAndFillRecord() 'প্রদর্শন সাব কল করা।
        End If
    End Sub
End Class
```

ঠিকানা ডাটাবেইজের পূর্ণাঙ্গ কোড তুলে ধরলাম। আমরা একে একে প্রতিটি সাব নিয়ে এবার বিস্তারিত ব্যাখ্যা তুলে ধরবো। তবে ইতোমধ্যে যেসব ব্যাপারে আলোচনা হয়ে গেছে তার পুনরাবৃত্তি করবো না অবশ্যই।
আমাদের কোডে সর্বমোট ৭টি সাবরুটিন ও ১টি ফাংশন আছে। প্রথমে এগুলোর নামের একটি তালিকা তুলে ধরছি।
১. Private Sub Button1_Click। ২. Private Sub EnableNavigationBtn। ৩. `Private Sub GetAndFillRecord`। ৪. `Private Sub btnFirst_Click`। ৫. `Private Sub btnLast_Click`। ৬. `Private Sub btnNext_Click`। ৭. `Private Sub btnPrev_Click`।
একটি মাত্র ফাংশন: `Private Function ValidateEntryNum`।

আমি শিক্ষার্থীদের সুবিধার জন্য মূল কোডে খুব বেশি করে কমেন্ট করেছি। এতে অনেক ব্যাখ্যা এসেছে। বাস্তবে যে কোনো কোডেই এরূপ ব্যাখ্যা রাখা জরুরী। পরবর্তীতে ডিবাগ [সঠিককরণ] ক্রিয়া অনেকটা সহজ হয়। আর মনে রাখা দরকার, কোনো কোনো কোড শত শত বা এমনকি হাজার হাজার লাইন পর্যন্ত হতে পারে। সুতরাং শিক্ষার্থীরা কমেন্ট করার অভ্যেস করে নেবেন। মনে রাখবেন যে কোনো লাইনের পূর্বে এই সিঙ্গুল কোট মার্ক ['] থাকলে সেটা কমেন্ট হয়ে যায়। কম্পাইল বা রান করার সময় এ লাইন উপেক্ষা করা হবে। যাক এবার ৭টি সাবরুটিন ও ১টি ফাংশনের ব্যাখ্যা একে একে তুলে ধরার প্রয়াস পাচ্ছি।
আমরা দু'টি গ্লোবাল ভেরিয়েবল ফর্মের ক্লাসের একেবারে উপরেই ডিক্লায়ার করেছি। আমাদের উদ্দেশ্য এগুলো যে কোনো সাবে প্রাপ্য হওয়া। এ গ্লোবালের প্রথমটির নাম: `RecNum`। আর দ্বিতীয়টির নাম: `NumRec`। প্রথমটিতে থাকবে ব্যবহারকারীর প্রবেশকৃত সংখ্যা- অর্থাৎ TextBox1 -এর সংখ্যাটি। আর দ্বিতীয়টিতে থাকবে আমাদের ঠিকানা ডাটাবেইজের টেবিলের রেকর্ড সংখ্যা। আমরা চাই রেকর্ড বৃদ্ধি কিংবা কমলেও যাতে এ্যাপলিকেশন কাজ করে। কোড ব্যাখ্যার পরই আমরা কিছু রেকর্ড বাড়াবো এবং প্রোগ্রাম রান করে দেখবো সবকিছু ঠিকমতো কাজ করছে কি না।

১. Private Sub Button1_Click: এই সাবরুটিন দ্বারা আমরা রেকর্ড প্রদর্শন করবো। এটা মূল সাবরুটিন

১৫তম পরিচ্ছেদ: ***Database Programming Made Simple***

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

GetAndFillRecord কল করে মাত্র।

২. Private Sub EnableNavigationBtn: এটার কাজ চারটি ন্যাভিগেশন বোতাম সক্রিয় করা। প্রথমে প্রোগ্রাম চলাকালে এই বোতামগুলো অক্রিয় রাখা থাকবে। এর কারণ হলে, তখনো ডাটাবেইজ খুলবে না। সুতরাং এগুলো সক্রিয় থাকার প্রয়োজন নেই। এই সাব ডাটাবেইজ খোলার পরই ডাকা হবে।

৩. Private Sub GetAndFillRecord: এটাই আমাদের মূল সাবরুটিন যার মাধ্যমে ডাটাবেইজ খোলা ও প্রদর্শিত হয়। এ সাব সম্পর্কে আগেই আমরা বেশ আলোচনা করেছি। তবে কিছু কোড নতুনভাবে যুক্ত হয়েছে। সুতরাং ওগুলো সম্পর্কে কিছু ব্যাখ্যার প্রয়োজন।

প্রথম লাইনটি দ্বারা আমরা গ্লোবাল ভেরিয়েবল RecNum -এ মান যুক্ত করেছি। লক্ষ করুন, এ মানটি হলো টেক্সবাক্স-১ এর মধ্যে যা আছে তাই। আর প্রথমে এ সংখ্যা হবে ০। অবশ্য পরবর্তিতে এটা ভিন্ন হতে পারে। এরপর যে কোডটি এখানে যুক্ত হয়েছে তাহলো con.Open() -এর পরের দু'টো লাইন এবং সবশেষে একটি লাইন। যথা:

```
'ক টি রেকর্ড আছে তা জানা। ভেরিয়েবল cmd একটি সিকুয়েল কমান্ড হোল্ড করবে।
Dim cmd As New OleDb.OleDbCommand("Select COUNT(*) FROM Tikana", con)
NumRec = cmd.ExecuteScalar() 'সিস্টেম ফাংশন দ্বারা রেকর্ড সংখ্যা আনা।
TextBox2.Text = NumRec 'রেকর্ড সংখ্যা টেক্স ফিল্ডে দেখানো।
```

লক্ষ করুন, উপরের কমেন্ট লাইনটি রেখে দিয়েছি যাতে শিক্ষার্থীরা ব্যাপারটি সহজে বুঝতে পারেন। প্রথম লাইনটি হলো **OleDb** এর একটি সিস্টেম অবজেক্ট ডিক্লারেশন। অবজেক্টটির একটি ইন্সট্যান্ট (বা কপি) **cmd** নামকরণে নতুনভাবে নিয়ে এসেছি। এর দুটি প্যারামিটার আছে: ১. সিকুয়েল কমান্ড "Select COUNT(*) FROM Tikana" এবং ২ con নামক কানেকশন অবজেক্ট। লক্ষ করুন প্রথম প্যারামিটারের কমান্ডে COUNT শব্দটি আছে। এর অর্থ হলো ঠিকানা টেবিলের মধ্যে ক'টি রেকর্ড আছে তা গণনা করা। দ্বিতীয় লাইন দ্বারা নির্দেশ দেওয়া হচ্ছে কমান্ড অবজেক্ট থেকে সংখ্যা নিয়ে এসে ভেরিয়েবল **NumRec** -তে রাখো। আর এ কাজের জন্য একটি সিকুয়েল নির্দেশনা তথা **ExecuteScalar** ব্যবহৃত হয়েছে। শিক্ষার্থীদের এ পর্যায়ে ঠিক কিভাবে এই কমান্ডগুলো কাজ করে তা জানার দরকার নেই। শুধুমাত্র এটাই শিখে নিতে হবে যে উপরোক্ত দু'টি লাইন দ্বারা যে কোনো ডাটাবেইজের উল্লেখিত টেবিলে ক'টি রো বা রেকর্ড আছে তা জানা যায়। আর মনে রাখার বিষয় যে, ডাটাবেইজ খোলা থাকাবস্থায় এই লাইনগুলো কার্যকরী হবে। এ দু'টি লাইনের একমাত্র উদ্দেশ্য হলো টেবিলে ক'টি রেকর্ড আছে তা জানা। শেষের লাইনটি দ্বারা রেকর্ড সংখ্যাটি টেক্সট্ বাক্স-২ এ প্রদর্শনের নির্দেশ দেওয়া হয়েছে। অবশ্য ফর্মে একটি লেবেল ও টেক্স বাক্স যুক্ত করতে হবে। এই সাবে আরো যা নতুনভাবে যুক্ত হয়েছে তাহলো:

```
IDText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(0) 'আইডি
```

এবং

```
EnableNavigationBtn() 'ডাটাবেইজ লোড হয়েছে তাই চারটি ন্যাভিগেশন বোতাম সক্রিয় করতে হবে|
```

যেহেতু আমাদের ডাটাবেইজে **ID** নামক একটি কলাম আছে তাই এটাও আইডি ফিল্ডে প্রদর্শনের প্রয়োজন। প্রথম লাইনে এ নির্দেশনাই দেওয়া হয়েছে। আর দ্বিতীয় লাইনের মাধ্যমে আমরা ন্যাভিগেশন বোতাম চারটে সক্রিয় করার জন্য সাবরুটিন কল করেছি।

৪. Private Sub btnFirst_Click। এ বোতামের মাধ্যমে আমরা প্রথম রেকর্ড দেখার ব্যবস্থা করেছি। লক্ষ করুন, অতি সহজে এ কাজটি আঞ্জাম দিতে আমরা প্রথমে টেক্সটবাক্স-১ এ ০ প্রবেশ করেছি এরপর কল করেছি আমাদের মূল সাবটি।

৫. Private Sub btnLast_Click। এ বোতামের মাধ্যমে আমরা সর্বশেষ রেকর্ডটি দেখাতে চাই। সুতরাং কোড প্রায় একই। এ ক্ষেত্রে অবশ্য ক'টি রেকর্ড আছে তা আমাদের জানা নেই। তবে, আমরা NumRec ভেরিয়েবলের মধ্যে এ সংখ্যাটি সংরক্ষণ করেছি। রেকর্ডের [বা রো এর] ইনডেক্স সিস্টেম কিন্তু ০- বেইজড্। সুতরাং সর্বশেষ রেকর্ডের ইনডেক্স হবে মোট রেকর্ড মাইনাস ১। এটা মনে রাখতে হবে।

৬. Private Sub btnNext_Click। এটা পরবর্তী রেকর্ডে ন্যাভিগেশনের কোড। তবে কোডটি কিছুটা জটিল। প্রথমে

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

জানতে হবে টেক্সটবাক্সে বর্তমানে যে সংখ্যাটি আছে তা ভেলিড কি না। কারণ, ব্যবহারকারী ইনভেলিড একটি সংখ্যা ঢুকিয়ে এ বোতামে ক্লিক করতে পারেন। ভেলিডিটি পরীক্ষার জন্য আমার `ValidateEntryNum()` টি কল করেছি। এটা কিভাবে কাজ করে একটু পরই বলবো। তবে এর ফলাফল সত্য বা মিথ্যা। প্রথম লাইনে আমরা বলছি রেজাল্ট যদি মিথ্যা না নয় [অর্থাৎ তা সত্য] তাহলে টেস্ট করে দেখতে হবে এ সংখ্যাটি রেকর্ডের সর্বশেষ ইনডেক্স সংখ্যা কি না। এটাও যদি না হয় তাহলেই আমরা পরবর্তী রেকর্ড দেখতে পারি। এ টেস্টের জন্য আমরা >= NumRec -এই কোডাংশ ব্যবহার করেছি। বলা হচ্ছে মোট রেকর্ড সংখ্যার সমান বা বেশি আছে কি? উভয়টির অর্থ হলো শেষ রেকর্ড প্রদর্শিত হচ্ছে। সুতরাং বিপ বাজিয়ে বের হও! অন্যথায়, পরবর্তী রেকর্ডে যাওয়া যায়।

এবার এই কোড লাইন: "`Dim entryNum As Integer = Val(TextBox1.Text) + 1 'টেক্স বাক্সের সংখ্যা ১ বাড়ানো।`" দ্বারা এনট্রি নাম্বার বাড়িয়েছি পরের রেকর্ডে যাওয়ার জন্য- কিন্তু! এখানেও আরেকটি টেস্টের দরকার। যদি নতুন নাম্বার রেকর্ডের সর্বশেষ ইনডেক সংখ্যা থেকেও বড়ো হয়ে যায়? সুতরাং দ্বিতীয় টেস্টও পাস করতে হবে। এবার পরের রেকর্ডে যাওয়ার নির্দেশ দেওয়া যায়। এ কাজে প্রথমে টেক্সট বাক্সে নতুন সংখ্যা দিয়েছি এরপর কল করেছি আমাদের মূল সাবটি।

৭. `Private Sub btnPrev_Click`। এটাও এইমাত্র ব্যাখ্যাত সাবের মতো কাজ করে। এ ক্ষেত্রে অবশ্য দেখতে হবে রেকর্ড সংখ্যাটি ০ থেকে নিচে হয়ে যায় কি না- এই যা।

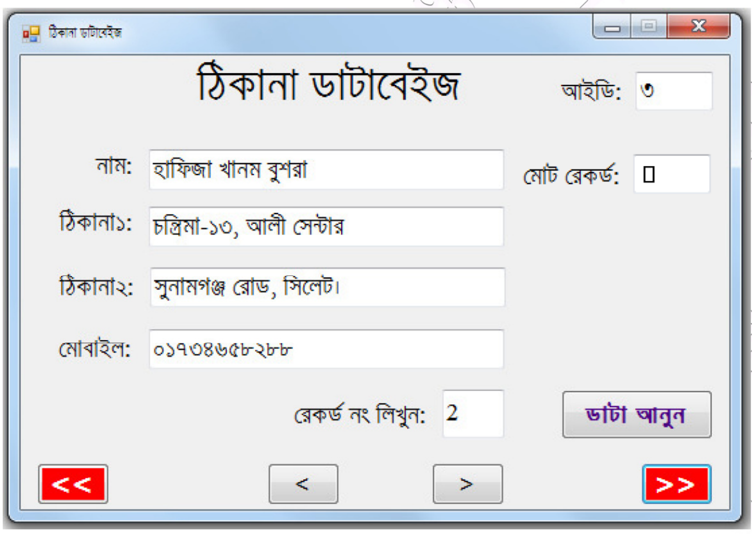
৮. `Private Function ValidateEntryNum`। সবশেষে এই ফাংশনটি কিভাবে কাজ করে তা একটু জেনে নিলে ভালো হয়। ফাংশনের নামেই বুঝা যায় এর কাজ হলো টেবিলের রো বা রেকর্ডের ইনডেক্স সংখ্যা সঠিক কি না যাচাই করে 'সত্য' বা 'মিথ্যা' তথা বুলিয়ান রেজাল্ট ফেরৎ পাঠানো। মনে রাখা প্রয়োজন, আমরা সর্বদাই টেক্সবাক্স-১ এর সংখ্যা নিয়ে এখানে কাজ করছি। এমনকি ন্যাভিগেশন বোতামগুলোর ক্রিয়ার সময়ও এই টেক্সবাক্সে ইনডেক্স সংখ্যা নিয়ে কোড রান করেছি। সুতরাং দেখতে হবে সংখ্যাটি ০ থেকে ইনডেক্স সংখ্যা পর্যন্ত যে কোনো একটি আছে কি না। না থাকলে ফাংশন একটি মেসেজ প্রদর্শন করে False রিটার্ন করবে। অন্যথায় রিটার্ন করবে True। লক্ষ করুন মেসেজ বাক্সে মোট ক'টি ইনডেক্স রেকর্ড আছে তা প্রদর্শিত হবে।

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

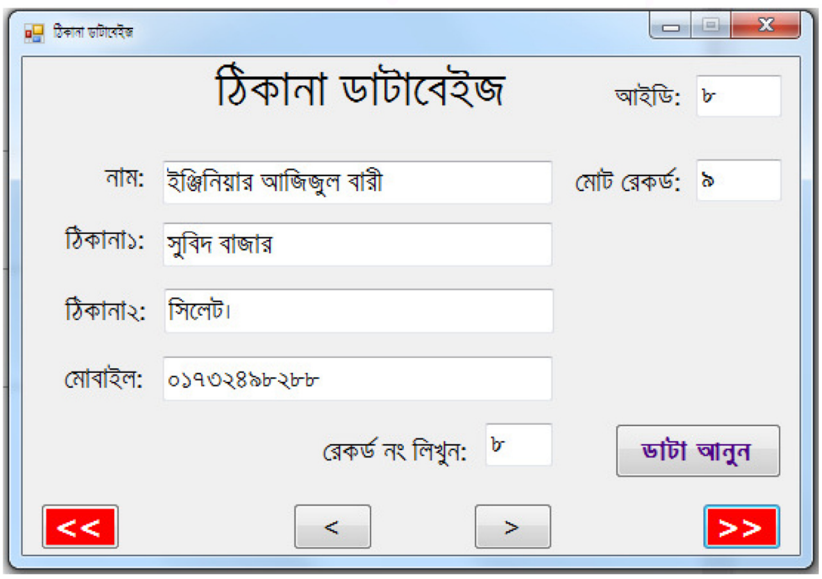
আলহামদুলিল্লাহ! ডাটাবেইজ প্রোগ্রামিংয়ের বেসিক কিছু কোর্ডিং সম্পর্কে আমরা অবগত হলাম। যে দু'টি ডাটাবেইজ এ্যাপলিকেশন আমরা তৈরি করেছি তা একেবারে সাধারণ নয়! এগুলো কাজে লাগানো যাবে। সবচেয়ে বড়ো কথা হলো আমাদের ডাটাবেইজ সম্পূর্ণ **বাংলা!**

এখানে একটি ব্যাপার উল্লেখ করতে হয়, প্রথমটির মতো দ্বিতীয়টিতে আমরা ন্যাভিগেশন কোডও দিয়েছি। তবে দু'টি বিষয় ছাড়া- মুছে ফেলা ও নতুন রেকর্ড তৈরি করা, এ দু'টো সুবিধা নেই। এগুলোর কোডিং বেশ জটিল তাই সেদিকে আপাতত যাচ্ছি না। এ দু'টো সুবিধাসহ ডাটাবেইজ চাইলে আমরা সহজেই ভিজুয়্যাল স্টুটিও ব্যবহার করে উইজার্ড দিয়ে ডাটাবেইজ বানাতে পারি- যেমনটি আমরা প্রথমে করেছি। তবে কোড দ্বারা কিভাবে সম্ভব তা-ও একটু পরে আমারা শিখে নেবো। এ পর্যন্ত যা হয়েছে তা পরীক্ষা করার পালা। আমরা ডাটাবেইজে প্রথম প্রোগ্রাম রান করে আরো কয়েকটি রো বা রেকর্ড যুক্ত করবো। এরপর এটি দ্বিতীয়টি দ্বারা প্রদর্শন করে দেখবো সব ঠিকমতো দেখা যায় কি না- কেমন?

টেস্ট



ঠিকানা ডাটাবেইজের রেকর্ড আগের মতো [৪টি] থাকাবস্থায় এ্যাপলিকেশন রান করে সবগুলো বোতাম পরীক্ষা করেছি। উপরে এর একটি চিত্র দেখুন।



ঠিকানা ডাটাবেইজের রেকর্ডে আরো কিছু [৫টি] নতুন ডাটা যুক্ত করার পর এ্যাপলিকেশন রান করে সবগুলো বোতাম পরীক্ষা করেছি। ডানের চিত্রে আমার সর্বশেষ [৯নং] রেকর্ডটি দেখতে পাচ্ছি।

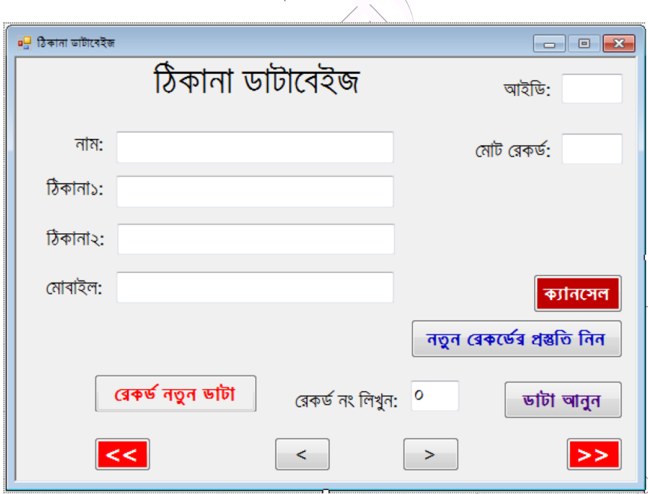
১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

ঠিকানা ডাটাবেইজে নতুন রেকর্ড সৃষ্টির কোড

আমরা এ পর্যন্ত যা করেছি তার দ্বারা রেডিমেইড কোনো এক্সেস ডাটাবেইজের তথ্যাদি প্রদর্শন সম্ভব। তবে সময় সময় তো নতুন রেকর্ড প্রবেশ করার প্রয়োজন দাঁড়াতে পারে। সুতরাং আসুন, দেখে নিই কিভাবে কোডের মাধ্যমে তা সম্ভব।

প্রথমত, একটি নতুন বোতাম ফর্মে যুক্ত করতে হবে। এর নামকরণ করুন □□□□□□□□ □□□ □□□□□! প্রয়োজনে আমরা ফর্মের সাইজ কিছুটা বড়ো করবো। বোমাতের নামকরণ করুন: btnNewRecord।



মনে রাখবেন, বাংলা লিখবেন বিজয়ের ইউনিকোডে। ইউনিকোড আনার শর্টকাট: Ctrl + Alt + V। একই শর্টকাট দ্বারা পুনরায় ইংরেজি লিখার কিবোর্ড আনা যায়। যাক, আমরা এবার নতুন বোতামের কোড লিখবো। তবে এর আগে আরো দু'টো বোতাম ফর্মে যুক্ত করা জরুরী।

প্রথমটি হলো একটি 'ক্যানসেল' বোতাম। নতুন ডাটা লেখার বোতামে ক্লিক করার পরও ব্যবহারকারী যাতে ইচ্ছা পরিবর্তন করতে পারেন সে সুযোগ তো থাকতে হবে। দ্বিতীয়টি হলো 'রেকর্ড নতুন ডাটা' নামকরণে একটি বোতাম। নতুন রেকর্ড ফিল্ডে প্রবেশ করার পর এ বোতামে ক্লিক করে ব্যবহারকারী আমাদের ডাটাবেইজ 'ঠিকানাডিবি' -তে তথ্য যুক্ত করবেন- মূলত নতুন একটি তথ্য সারি যোগ হবে। আমরা বোতামগুলো যুক্ত করে একটি ইউই [User Iterface] তৈরি করেছি। উপরের চিত্রটি দেখুন। লক্ষ করুন, **'ক্যানসেল'** বোতামের (Name) Propery হবে: btnCancel। এর Visible Property হবে: False। **'রেকর্ড নতুন ডাটা'** বোতামের (Name) Propery হবে: btnRecord। এটার Enabled Property হবে: False। এবার আমরা কোডগুলোর প্রতি দৃষ্টি দিতে পারি।

নতুন রেকর্ড যুক্ত করার কোডটি কিছুটা জটিল। সুতরাং খুব সতর্কসহ ব্যাখ্যাটুকু পাঠ করে নেবেন। অবশ্য ইতোমধ্যে যেসব সাবরুটিন সম্পর্কে বলেছি এবং সেগুলো এখনও পরিবর্তিত আছে ওগুলোর ওপর ব্যাখ্যর পুনরাবৃত্তি করবো না। আমরা পুরো প্রোগ্রাম শেষে সমস্ত কোড আবার একসাথে উল্লেখ করবো- অবশ্যই। প্রথমেই যে কাজটি করা জরুরী তাহলো কিছু ডাইমেনশন [Dim] ডিক্লারেশন সর্বত্র প্রাপ্তিযোগ্য করতে হবে। এজন্য অবশ্য ফর্মের ক্লাস ডিক্লারেশনের পরই এসব কোড লিখতে হবে। অর্থাৎ গ্লোবাল ডিক্লারেশন হবে নিম্নরূপ:

```
'গ্লোবাল ডিক্লারেশন
    Dim RecNum As Integer = 0 'এটা হোল্ড করবে টেক্সট্ বাক্সের সংখ্যাটি যার দ্বারা সঠিক রো এর তথ্য আনা হবে।
    Dim NumRec As Integer 'এটা হোল্ড করবে টেবিলের মোট রেকর্ড নাম্বার।
    Dim KeepingIndex As Integer 'নতুন রেকর্ড তৈরিতে প্রদর্শিত রেকর্ডের ইনডেক্স নাম্বার সংরক্ষণ
    Dim ds As New DataSet 'একটি ডাটাসেট অবজেক্ট হোল্ড করে।
    Dim da As OleDb.OleDbDataAdapter 'একটি এডাপটার অবজেক্ট হোল্ড করে।
    Dim con As New OleDb.OleDbConnection 'ভেরিয়েবল con একটি যুক্তকরণ অবজেক্ট হোল্ড করবে।
    Dim dbProvider As String 'ভেরিয়েবল dbProvider একটি বিশেষ স্ট্রিং হোল্ড করবে।
    Dim dbSource As String 'ভেরিয়েবল dbSource একটি স্ট্রিং হোল্ড করবে।
    Dim FullDatabasePath As String 'ডাটাবেইজের পূর্ণাঙ্গ প্যাথনাম হোল্ড করবে।
    Dim Sql As String 'একটি সিকুয়েল স্ট্রিং হোল্ড করবে।
```

সবগুলো ভেরিয়েবলের ডিক্লারেন শেষে কমেন্ট লাইনে বুঝিয়ে দিয়েছি। সুতরাং এখানে আর কিছু যোগ করার নেই- শুধু একটি ছাড়া। এ ব্যাপারটি হলো ভেরিয়েবলগুলোর ডাটা টাইপ। মোট তিনটি ইন্টেজার [আস্ত সংখ্যা], চারটি স্ট্রিং [লেখা] এবং তিনটি অবজেক্ট টাইপ ডিক্লারেশন আছে। এগুলো কিভাবে কাজ করে তা পরবর্তিতে বুঝিয়ে বলছি।

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

এবার প্রথম সাবরুটিনটি এখানে তুলে ধরছি:

```
' ডাটা আনুন বোমাতের ক্লিক ইভেন্ট। ব্যবহারকারী প্রথমেই এ বোতামে ক্লিক করবেন।
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    GetAndFillRecord() 'তথ্য আনার সাব কল।
    btnNewRecord.Enabled = True 'এই লাইটটি দ্বারা রেকর্ড নতুন ডাটা বোতামটি সক্রিয় করা হয়েছে।
End Sub
```

আমাদের ঠিকানা ডাটাবেইজ রান করলেই ঠিকানা টেবিল দেখতে চাইলে উপরোক্ত বোতামে ক্লিক করা ছাড়া বিকল্প কিছু নেই। সুতরাং এটাই প্রথমে একশনে যাবে। আমরা সুবিধার জন্য তথ্য এনে প্রদর্শনের কোড ব্লকটি আলাদা একটি সাবরুটিনে করেছি। প্রথম লাইন সেই সাবরুটিন কল করে ও সাথে সাথে নতুন রেকর্ড ঢুকানোর জন্য এর প্রোপার্টি সক্রিয় করে দেয়। ব্যস! এ বোতামের কাজ এটুকুই।

```
'রেকর্ড থেকে তথ্য এনে ফিল্ডে দেখানো হলো এ সাবের কাজ
Private Sub GetAndFillRecord()
    RecNum = Val(TextBox1.Text) 'ভেরিয়েবলে টেক্সবাক্সের সংখ্যা। প্রথমবার এর মান ০।
    dbProvider = "Provider=Microsoft.ACE.OLEDB.12.0;" 'সিস্টেমের বিশেষ একটি স্ট্রিং।
    FullDatabasePath = "C:\Users\htci\Desktop\LearningDotNet\MyDatabase\TikanaBoi.accdb"
    dbSource = "Data Source=" & FullDatabasePath 'dbSource এর মধ্যে এখন পূর্ণাঙ্গ স্ট্রিং থাকবে।
    If con.State <> ConnectionState.Open Then 'কানেকশন স্ট্রিং আছে কি না দেখতে হবে।
        con.ConnectionString = dbProvider & dbSource 'কানেকশন স্ট্রিং পূর্ণ হলো।
        con.Open()
    End If
    Dim cmd As New OleDb.OleDbCommand("Select COUNT(*) FROM Tikana", con)
    NumRec = cmd.ExecuteScalar() 'সিস্টেম ফাংশন দ্বারা রেকর্ড সংখ্যা আনা।
    'ফাংশন কল করে রেকর্ড সংখ্যা সঠিক আছে কি না দেখা।
    If ValidateEntryNum() = False Then
        con.Close() 'ভুল হলে ডাটাবেইজ বন্ধ করে এক্সিট করা
        Exit Sub
    End If
    Sql = "SELECT * FROM Tikana" 'সিকুয়েল স্ট্রিং স্টোর করা।
    'ডাটা এডাপটারের নিকট সিকুয়েল স্ট্রিং ও কানেকশন অবজেক্ট পাস করে দেওয়া।
    da = New OleDb.OleDbDataAdapter(Sql, con)
    da.Fill(ds, "TikanaDB") 'দ্বিতীয় প্যারামিটার যে কোনো কিছু হতে পারে।
    IDText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(0) 'আইডি
    NameText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(1) 'নাম
    AddressText1.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(2) 'ঠিকান-১
    AddressText2.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(3) 'ঠিকানা-২
    MobileText.Text = ds.Tables("TikanaDB").Rows(RecNum).Item(4) 'মোবাইল
    EnableNavigationBtn() 'ডাটাবেইজ লোড হয়েছে তাই চারটি ন্যাভিগেশন বোতাম সক্রিয় করতে হবে।
    TextBox2.Text = NumRec 'রেকর্ড সংখ্যা টেক্সট ফিল্ডে দেখানো।
End Sub
```

লক্ষ করুন, উপরোক্ত মূল সাবে বেশ রদবদল হয়েছে। প্রথমত, একটি মাত্র Dim স্টেটমেন্ট আছে। বাকিগুলো গ্লোবেল হিসাবে পূর্বেই ডিক্লায়ার করা হয়েছে। দ্বিতীয়ত একটি অতিরিক্ত IF--End If ব্লক যুক্ত হয়েছে। এছাড়া শেষের দিকের একটি লাইন এখানে নেই। সেটা হলো: con.Close()। এর কারণ হলো, এ পর্যায়ে আমরা ডাটাবেইজের সঙ্গে সংযুক্তি ক্লোজ করতে চাই না। পরিবর্তিত কোডগুলোর ব্যখ্যা পরের পৃষ্ঠায় এসেছে।

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

পরিবর্তিত কোড

মূল `GetAndFillRecord()` নামক সাবরুটিনের কোডে নিম্নলিখিত পরিবর্তন এসেছে:

১.
```
If con.State <> ConnectionState.Open Then 'কানেকশন স্ট্রিং আছে কি না দেখকে হবে।
    con.ConnectionString = dbProvider & dbSource 'কানেকশন স্ট্রিং পূর্ণ হলো।
    con.Open()
End If
```

এ ব্লকটির মাধ্যমে টেস্ট করা হয়, ডাটাবেইজের সঙ্গে কানেকশন খোলা আছে কি না। থাকলে কিন্তু পুনরায় খোলার নির্দেশ আসলে কম্পাইলারে অভিযোগ করবে। আমরা একাধিকবার এই সাবটি কোড রানিংয়ের সময় ডাকবো- তাই এ টেস্ট। বুঝতেই পারছেন, খোলা না থাকলে দ্বিতীয় লাইন কানেকশন স্ট্রিং প্রভাইড করবে ও তৃতীয় লাইন ডাটাবেইজের সঙ্গে কানেকশন সৃষ্টি করবে। এই সাবে এটুকুই পরিবর্তিত হয়েছে। শেষের লাইনটি যে কেটে দেওয়া হয়েছে তা-তো ইতোমধ্যে কারণসহ উল্লেখ করেছি।

২.
```
'new record button click event sub
 Private Sub btnNewRecord_Click(sender As Object, e As EventArgs) Handles btnNewRecord.Click
     If isAllRecEntered() = False Then 'not a new record
         Beep()
         Exit Sub
     End If
     Dim RetVal As Integer = MsgBox("নতুন রেকর্ড লিখবেন?", MsgBoxStyle.YesNo)
     If RetVal = 7 Then Exit Sub
     KeepingIndex = Val(TextBox1.Text) 'keep the old index number incase cancelled
     IDText.Text = NumRec 'num of Record always 1 more then o index record number. No editing
     TextBox2.Text = NumRec + 1 'show number of record will gone up by one. No editing
     DeleteTextFromFields() 'call sub to delete fields texts
     DisableNavigationBtn() 'call sub to disable navigation buttons.
     btnRecord.Enabled = True
 End Sub
```

এই সাবটি সম্পূর্ণ নতুন। এটা 'নতুন রেকর্ডের প্রস্তুতি নিন' বোতামের সাবরুটিন। এতে ক্লিক করলেই প্রথমে একটি টেস্ট করতে হবে। প্রথমেই দেখতে হবে কোনো ফিল্ডে টেক্সট আছে কি না। আমরা এখানে একটি ফাংশন কল করেছি। এটির কাজ হলো সকল টেক্সট ফিল্ডে লেখা থাকলে 'সত্য' এবং না থাকলে 'মিথ্যা' রিটার্ন করা। যদি 'মিথ্যা' রিটার্ন করে তাহলে বুঝা যাবে, ইতোমধ্যে আমরা নতুন রেকর্ডের জন্য নির্দেশ করে নিয়েছি। অন্যকথায় বোতামে ক্লিক করে শূন্য ফিল্ডগুলো নিয়ে এসেছি। পরের লাইনটি সতর্কবার্তা। নতুন রেকর্ড সত্যিই লিখা হবে কি না- তা পুনরায় জেনে নেওয়া। অবশ্য এখানেও ব্যবহারকারী ভুল করলে, পরবর্তিতে আরেকটি ক্যানসেল বোতাম আছে। সেটিতে ক্লিক করে মূল রেকর্ডে ফিরে যেতেও পারবেন। লক্ষ করুন, মেসেজ বাক্সে 'No' ক্লিক করলে রিটার্ন ভ্যালু হয় ৭। এজন্যই আমরা পরের লাইনে বলেছি RetVal ৭ হলে এক্সিট করার জন্য। জেনে রাখুন, ব্যবহারকারী ডায়লগে 'Yes' ক্লিক করলে রেজাল্ট হবে ৬।

এখন, মনে করুন ব্যবহারকারী ইয়েস চয়েজ করলেন। কিন্তু পরে আবার ক্যানসেলে ক্লিক করলেন, তখন তো আগে প্রদর্শিত তথ্যগুলো পুনরায় প্রদর্শন করতে হবে। তথ্যাদি প্রদর্শিত হয় টেবিলের [0-Based] ইনডেক্সের মাধ্যমে। আমাদের `TextBox1` -এ এই ইনডেক্স নাম্বারটি আছে। সুতরাং `NumRec` নামক আমাদের গ্লোবাল ভেরিয়েবলে বাক্সের সংখ্যাটি সংরক্ষণের ব্যবস্থা করা হয়েছে। নতুন রেকর্ড প্রবেশের সকল ফিল্ড খালি হবে, তাই এ সংখ্যাটি থাকলে ক্যানসেল বোতামে ক্লিক করলেই পুনরায় আগের রেকর্ডটি ডাটাবেইজ থেকে প্রদর্শন নিশ্চিত করা যাবে। পরের লাইন দ্বারা রেকর্ড সংখ্যাটি [আইডি] ফিল্ডে প্রদর্শিত হয়েছে। টেক্সট বাক্স-২ এ প্রদর্শিত হয়েছে [নতুন রেকর্ডসহ] রেকর্ড সংখ্যা।

এবার আমরা কল করেছি `DeleteTextFromFields()` সাবরুটিন। এর কাজ হলো চারটি এনট্রি ফিল্ড মুছে দেওয়া। পরের লাইনও `DisableNavigationBtn()` নামক সাবরুটিন কল। এটার কাজ সকল ন্যাভিগেশন বোতাম অক্রিয় করা। সব শেষে নির্দেশ দিচ্ছি, 'রেকর্ড নতুন ডাটা' বোতামটি সক্রিয় করে দিতে- এরই মাধ্যমে আমরা নতুন রেকর্ড ঢুকাবো।

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

৩.
```
'disables all four navigation btns
Public Sub DisableNavigationBtn()
    btnLast.Enabled = False : btnFirst.Enabled = False
    btnNext.Enabled = False : btnPrev.Enabled = False
    btnCancel.Visible = True
End Sub
```

এই সাবরুটিনের কাজ হলো চারটি ন্যাভিগেশন বোতাম অক্রিয় ও শুধুমাত্র ক্যানসেল বোতামটি প্রদর্শন করা। লক্ষ করুন, কলোন (:) এর ব্যবহার। একই লাইনে একাধিক নির্দেশ দিতে চাইলে এভাবে কলোন ব্যবহার করতে হয়। এই সাবরুটিনটি নতুন রেকর্ডের প্রস্তুতির অংশ।

৪.
```
'deletes five text fields and makes 4 of them editable
Public Sub DeleteTextFromFields()
    NameText.Text = "" : AddressText1.Text = ""
    AddressText2.Text = "" : MobileText.Text = ""
    NameText.ReadOnly = False : AddressText1.ReadOnly = False
    AddressText2.ReadOnly = False : MobileText.ReadOnly = False
    TextBox1.Text = "" 'we dont't want user to edit that field
    Button1.Enabled = False 'disable get record btn
End Sub
```

এটাও নতুন রেকর্ড প্রবেশ করার জন্য প্রস্তুতির অংশ। লক্ষ করুন, ডিজাইন টাইমে আমরা চারটি তথ্যফিল্ড অক্রিয় করে রেখেছিলাম। কারণ, আমরা চাইনি ব্যবহারকারী অসতর্কতাহেতু কোনো রেকর্ডে পরির্তন আনেন। তবে এবার তো নতুন রেকর্ড লিখা হবে তাই প্রথমে সবগুলে মুছে ফেলার পর এগুলো সক্রিয় করেছি। কিন্তু TextBox1 -এ কিছু লেখা আমরা চাই না। তাই একে মুছার পরও অক্রিয় রেখেছি। সব শেষের লাইন দ্বারা 'ডাটা আনুন' বোতামটি অক্রিয় করা হয়েছে।

৫.
```
'this sub locks all 4 flds
Public Sub MakeFourFldReadOnly()
    NameText.ReadOnly = True : AddressText1.ReadOnly = True
    AddressText2.ReadOnly = True : MobileText.ReadOnly = True
End Sub
```

রেকর্ড ঢুকানো কিংবা ক্যানসেল হওয়ার পর চারটি তথ্যফিল্ডকে পুনরায় অক্রিয় করতে হবে- যাতেকরে ব্যবহারকারী আর কিছু লিখতে না পারেন। এ কাজটিই আঞ্জাম দেয় উপরোক্ত সাবটি।

৬.
```
'retuns true if all 4 flds contains text, ready for record validation
Function isAllRecEntered()
    If NameText.Text <> "" And AddressText1.Text <> "" _
       And AddressText2.Text <> "" And MobileText.Text <> "" Then
        Return True
    End If
    Return False
End Function
```

ইতোমধ্যে এ ফাংশনটির কথা উল্লেখ করেছি। এর কাজ হলো টেক্স চারটিতে লেখা আছে কি না সে বুলিয়ান তথ্য রিটার্ন করা।

৭.
```
'btnCancel event code, simply show the last record, disable itsef, enable get rec btn
Private Sub btnCancel_Click(sender As Object, e As EventArgs) Handles btnCancel.Click
    TextBox1.Text = KeepingIndex 'এটার মধ্যে আগে প্রদর্শিত রেকর্ড ইনডেক্স আছে।
    GetAndFillRecord() 'রেকর্ড প্রদর্শনের সাবরুটিন কল করা।
    MakeFourFldReadOnly() 'ফিল্ড চারটে রীড-অনলি বানানো।
    btnCancel.Visible = False : Button1.Enabled = True 'enable get record btn
    btnRecord.Enabled = False : btnNewRecord.Enabled = True
End Sub
```

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

আগের পৃষ্ঠার ৭ নাম্বারে প্রদর্শিত 'ক্যানসেল' বোতামের সাবরুটিন দ্বারা ইতোমধ্যে যে রেকর্ডটি টেক্স ফিল্ডগুলোতে ছিলো তা পুনরায় দৃশ্যমান করা হয়েছে। এটা প্রথমে ইনডেক্স সংখ্যা টেক্সবাক্স-১ এ লিখে দেয়। এরপর কল করে একে একে ইতোমধ্যে আলোচিত [GetAndFillRecord() এবং MakeFourFldReadOnly() নামক] দু'টি সাবরুটিন। প্রথমটি তো আমাদের মূল সাব- যার কাজ রেকর্ড প্রদর্শন করা আর দ্বিতীয়টির কাজ হলো চারটি ফিল্ডকে লেখার জন্য অক্রিয় করে দেওয়া।

৮.
```
    'record Button
Private Sub btnRecord_Click(sender As Object, e As EventArgs) Handles btnRecord.Click
    Dim MsgStr As Integer = MsgBox("সতর্কবার্তা! সবকিছু সঠিক মতো লিখেছেন তো?", MsgBoxStyle.YesNo)
    'MsgBox(MsgStr)
    If MsgStr = 7 Then
        Exit Sub
    End If
    AddRows()
End Sub
```

এই btnRecord ['রেকর্ড নতুন ডাটা'] বোতামের ক্লিক ইভেন্ট কল করে সাবরুটিন 'AddRows()'। এটা ডাকার পূর্বেও একটি সতর্কবার্তা ডায়লগ দেখানোর ব্যবস্থা করেছি। কারণ ব্যবহারকারী ভুল করতে পারেন তথ্য প্রবেশে- কিংবা তিনি নতুন রেকর্ড এ মুমূর্তে ঢুকানোর ক্রিয়া ক্যানসেলও করার ইচ্ছা রাখতে পারেন। তাকে এই সুযোগটুকু তো দিতে হবে না? প্রথম চার লাইন এ সুযোগ দিয়েছে।

৯.
```
Private Sub AddRows()
    If isAllRecEntered() = False Then
        MsgBox("সব রেকর্ড সঠিকভাবে ঢুকিয়েছেন কি না দেখুন! তথ্য রেকর্ড হয় নি।")
       Exit Sub
    End If
    EntryToRaw() 'রেকর্ড ঢুকানোর সাবরুটিন কল করা।
GetAndFillRecord() 'রেকর্ড প্রদর্শনের সাবরুটিন কল করা।
MakeFourFldReadOnly() 'ফিল্ড চারটে রীড-অনলি বানানো।
    btnCancel.Visible = False
    Button1.Enabled = True 'enable get record btn
    btnRecord.Enabled = False
    btnNewRecord.Enabled = True
End Sub
```

এই সাবরুটিনইটিই হলো রেকর্ড টেবিলে যোগ করার পূর্ব প্রস্তুতির সর্বশেষ ধাপ। EntryToRaw() নামক সর্বশেষ সাবরুটিন কলের পূর্বে একটি টেস্ট করা হয়। সকল ফিল্ডে টেক্সট প্রবেশ করা হয়েছে কিনা তা জানতে হবে। যদি সকল বাক্সে টেক্সট না থাকে, তাহলে মেসেজ বাক্সের মাধ্যমে সতর্কবাণী প্রদর্শন করে বেরিয়ে যেতে হবে সাবরুটিন থেকে। এরপর [সবটি ঠিকঠাক থাকলে] প্রথমে EntryToRaw() সাবরুটিন কল করে নতুন রেকর্ড তৈরি করা হয়। এরপর রেকর্ড প্রদর্শনের মূল সাবরুটিন কল করে আগের প্রদর্শিত তথ্যাদি টেক্সট বাক্সগুলোতে পুনরায় দেখানো হচ্ছে। এর পরের সাবরুটিন কল করে সকল টেক্স বাক্স রিড-অনলি করতে হবে। সবশেষে তিনটি বোতামের ক্রিয়া পরিবর্তন করা জরুরী: ক্যানসেল বোতাম গোপন করা; রেকর্ড আনার বোতাম সক্রিয় করা; রেকর্ড করার বোতাম নিষ্ক্রিয় করা; এবং নতুন রেকর্ডের প্রস্তুতি-বোতাম আবার সক্রিয় করা।

এ হচ্ছে সর্বশেষ ডাটা এনট্রি সাবের পূর্ব পর্যন্ত সাবগুলোর ক্রিয়ার বর্ণনা। আমরা এবার পরের পৃষ্ঠায় এক্সেস টেবিলে তথ্যাদি রেকর্ড করার ক্রিয়াটির পূর্ণাঙ্গ ব্যাখ্যা তুলে ধরবো।

১৫তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
'Code for data entry into access table.
   Private Sub EntryToRaw()
       Dim cb As New OleDb.OleDbCommandBuilder(da) 'commandBuilder Object creation
       Dim dsNewRow As DataRow 'new data row creation variable
       'notice here, ID column as number will be added automatically [AutoNumber]
       dsNewRow = ds.Tables("TikanaDB").NewRow() 'adding a row into TikanaDB
       dsNewRow.Item(1) = NameText.Text 'name text into 2nd column cell
       dsNewRow.Item(2) = AddressText1.Text 'Address text into 3rd column cell
       dsNewRow.Item(3) = AddressText2.Text 'address text2 into 4th column cell
       dsNewRow.Item(4) = MobileText.Text 'mobile number [as text] into 5th column text
       ds.Tables("TikanaDB").Rows.Add(dsNewRow) 'actual row of data in dataset
       da.Update(ds, "TikanaDB") 'the row of data in ds goes to the created row
       con.Close() 'now close the connection
       MsgBox("ডাটাবেইজে নতুন রেকর্ড সফলভাবে প্রবেশ করা হয়েছে।") 'show success message
   End Sub
```

সাবের প্রথম লাইন: কমান্ড বিল্ডার সিস্টেম অবজেক্টের নতুন ইন্সট্যান্ট ডিক্লারেশন (আমাদের নামকরণ: cb)। এতে একটি প্যারামিটার আছে- তাহলো ডাটা এডাপটার [আগেই ডিক্লায়ার করা এই এডাপটারের আমাদের কর্তৃক নামকরণ: da)।

দ্বিতীয় লাইন: একটি ডাটারো টাইপ ডিক্লারেশন [টেবিলে নতুন তথ্যলাইন সৃষ্টির জন্য এটি, আমাদের নামকরণ: dsNewRow]

তৃতীয় লাইন: উক্ত ডাটাটাইপ ভেরিয়েবলে ঠিকানা ডাটাবেইজের টেবিলে একটি নতুন রো [বা তথ্যলাইন] সৃষ্টি করে dsNewRow -তে সংরক্ষণের নির্দেশ। লক্ষ করুন, ds.Tables নির্দেশনার একটি প্যারামিটার হলো আমাদের TikanaDB। এখানে ds হলো পূর্বেই সৃষ্ট আমাদের DataSet। সবশেষে .NewRow() দ্বারা পয়েন্ট করা হচ্ছে নতুন রো সৃষ্টির ফাংশনের দিকে। শিক্ষার্থীদের প্রতি অনুরোধ, এসব সিস্টেম নির্দেশনা কিভাবে কাজ করে, সে সম্পর্কে কোনো ভাবনার প্রয়োজন নেই। এতো গভীরে যাওয়া আমাদের কাজ নয়- বরং ওসব কাজ সফটওয়্যার বিজ্ঞানীদের। অর্থাৎ মাইক্রসফটের ইঞ্জিনিয়ারদের। আমরা তাদের আবিষ্কার ও উদ্ভাবনগুলো কাজে লাগাবো কিভাবে- সেটাই জানতে হবে।

চতুর্থ থেকে সপ্তম লাইন: এই লাইনগুলো দ্বারা ব্যবহারকারী কর্তৃক চারটি ফিল্ডে প্রবেশকৃত টেক্সট ইনডেক্স নির্দেশনার মাধ্যমে ডাটাসেটে সংরক্ষণ করেছি। লক্ষ করুন, আইডি সংখ্যা সংরক্ষণের প্রয়োজন নেই। এটা এক্সেস নিজেই বানাবে। কারণ আমরা এই সেলকে AutoNumber ডটাটাইপ হিসাবে তৈরি করেছি। এ পর্যন্ত সকল নির্দেশনা ডাটাসেট এর মধ্যে ds নামকরণে মেমোরিতে সংরক্ষণ করেছি। সবকিছু প্রস্তুত হয়ে গেছে। এবার সত্যিকার অর্থে টেবিলের মধ্যে একটি তথ্যলাইন যোগ করার নির্দেশ দিতে হবে।

অষ্টম লাইন: এই লাইনটি কিভাবে কাজ করে তা একটু বুঝার দরকার। মনে রাখবেন, প্রতিটি ডট [.] হলো পয়েন্টার ইঙ্গিত। প্রথমে ডাটাসেট [ds] পয়েন্ট করছে আমাদের ঠিকানা টেবিলের দিকে (.TikanaDB)। এরপর পয়েন্ট করছে রো বা তথ্যলাইনের দিকে [.Raw]। সবশেষে পয়েন্ট করছে নতুন রো সৃষ্টির দিকে [.Add(dsNewRow)]। এ লাইনের কোড রান হওয়ার পর আমাদের এক্সেস টেবিলে একটি নতুন তথ্যলাইন বাস্তবে যোগ হবে। এতে শুধুমাত্র আইডি ফিল্ডে একটি ইউনিক [আলাদা] আইডি সংখ্যা থাকবে। বাকী চারটি সেল এখনও শূন্য। পরের লাইনের নির্দেশ দ্বারা এগুলো পূরণ করা হবে।

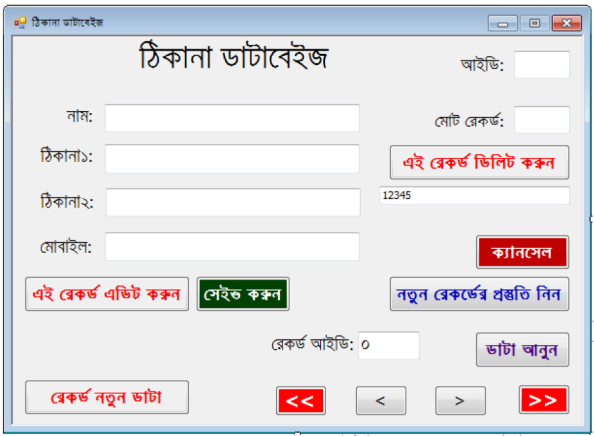
নবম লাইন: ds.Update একটি ফাংশন। এতে আছে দু'টি প্যারামিটার `(ds, "TikanaDB")`। প্রথমটি আমাদের সৃষ্ট ডেটাসেট যেখানে সংরক্ষিত আছে চারটি টেক্সট বাক্সের তথ্যাদি। দ্বিতীয় প্যারামিটার হলো আমাদের এক্সেস ডাটাবেইজের নাম। ব্যস! তথ্যাদি সেলের মধ্যে চলে যাবে।

দশম লাইন: কাজ শেষ। এবার কানেকশন বন্ধ করার পালা। এই লাইনে তা-ই নির্দেশ করা হয়েছে।

শেষ লাইন: ব্যবহারকারীকে তথ্য দেওয়া হচ্ছে- নতুন তথ্যলাইন তৈরি হয়েছে সফলভাবে। এখন প্রোগ্রাম বন্ধ করে পুনরায় চালালে নতুন রেকর্ডও দেখাবে।

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

আমাদের ঠিকানা ডাটাবেইজের কাজ এখনো পুরোদমে শেষ হয় নি। যে কোনো তথ্য আপডেট করা ও মুছে দেওয়ার সুযোগ তো থাকতেই হবে। আমরা এখনই এ কাজগুলোর জন্য কোড লিখে নেবো। প্রথমে ফর্মের মধ্যে আরো তিনটি বোতাম যুক্ত করতে হবে। এর প্রথমটির টেক্সট্ প্রোপার্টি হবে 'এই রেকর্ড ডিলিট করুন'; নামকরণ: btnDelete। দ্বিতীয়টির টেক্সট্ প্রোপার্টি হবে 'এই রেকর্ড এডিট করুন'; নামকরণ: btnEdit। তৃতীয়টির টেক্সট্ প্রোপার্টি হবে 'সেইভ করুন'; নামকরণ: btnUpdate। সবগুলো বোতাম সুন্দরভাবে সাজিয়ে রাখুন। দেখুন বায়ের চিত্রটি। এবার কোডিং করতে হবে।



১. 'এই রেকর্ড ডিলিট করুন' বোতামের কোড: নতুন রেকর্ড করার কোড থেকে এ বোতামের কোড অনেকটা সহজ। দৃশ্যমান রেকর্ডটি ডিলিট হবে। বোতামের কোড প্রথমে দেখে নেই- এরপর ব্যাখ্যা আসবে।

```
'delete button's code
Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles btnDelete.Click
    ' first confirmation
    Dim ResP As Integer = MsgBox("আপনি কি সত্যিই এই রেকর্ডটি মুছে ফেলবেন?", MsgBoxStyle.YesNo)
    If ResP = 7 Then
        Exit Sub
    End If
    Dim RpT As String = InputBox("Enter Delete Record Password.")
        If ResP <> DeleteIdFld.Text Then
            MsgBox("Sorry, wrong Password! Record can not be deleted!", MsgBoxStyle.Critical)
            Exit Sub
        End If
    DeleteRow() 'call deleting sub
    If RecNum >= 0 Then
        GetAndFillRecord() 'রেকর্ড প্রদর্শনের সাবরুটিন কল করা।
    End If
End Sub
```

প্রথমত, ব্যবহারকারীর জন্য একটি সেফটি নেটের প্রয়োজন। তিনি ভুলবশত, এ বোতামে ক্লিক করতে পারেন। সুতরাং একটি মেসেজ বাক্সে আবারো জিজ্ঞেস করা হচ্ছে, সত্যিই তিনি এই রেকর্ডটি মুছে দিতে চান কি না? অবশ্য আরো সতর্কতার জন্য একটি গোপন টেক্সটফিল্ডে নাম্বার বা পাস্ওয়ার্ড রাখা যায়। আমরা সেটাও করবো। ফর্মের মধ্যে একটি টেক্সট্ বাক্স নিয়ে আসুন, এর নামকরণ করুন: 'DeleteIdFld'। এর Visible Property হবে False। আইডি যা ইচ্ছা লিখতে পারেন। সহজ করতে যেয়ে আমরা বাক্সের Text Property -তে লিখবো: ১২৩৪৫। অবশ্য আমরা চাইলে সংখ্যাটি সরাসরি কোডে ঢুকিয়ে টেস্ট করতে পারতাম। তবে সুবিধার জন্য ফিল্ডে রাখাই ভালো। যাক উভয় কন্ডিশন পাস হওয়ার পরই আমরা **DeleteRow()** নামক সাবরুটিনটি কল করেছি। এটির কাজ হলো, যে রেকর্ড এ মুহূর্তে দৃশ্যমান আছে তা টেবিল থেকে মুছে ফেলা।
ডিলিট হওয়ার পর দেখতে হবে আর কোনো রেকর্ড আছে কিনা। এটা নিশ্চিত করতেই পরের If RecNum >= 0 ব্লকটি। যদি রেকর্ড থেকে থাকে, তখনই কল করেছি রেকর্ড পুনরায় ফিল্ডে নিয়ে আসার জন্য- আর তা হবে, এই মাত্র মুছে যাওয়া রেকর্ডের আগেরটি। এবার দেখে নিই রেকর্ড মুছে ফেলার সাবরুটিনটি। পরের পৃষ্ঠায় দেখুন।

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
'row delete codes
Sub DeleteRow()
    If NumRec = 0 Then 'incase there aren't any row of data
        MsgBox("টেবিলে তো আর কোনো রেকর্ড নেই!")
        Exit Sub
    End If
    Dim cb As New OleDb.OleDbCommandBuilder(da) 'commandBuilder Object creation
    ds.Tables("TikanaDB").Rows(RecNum).Delete() 'actual delete row command
    da.Update(ds, "TikanaDB") 'update database
    TextBox1.Text = RecNum - 1
    MsgBox("এই রেকর্ড মুছে গেছে।") 'show success message
    con.Close() 'close connection
End Sub
```

প্রথমে টেস্ট। যা কাটা হবে তা আছে কি? না থাকলে মেসেজ প্রদর্শন করে সাব থেকে বের হতে হবে। মনে রাখবেন, NumRec একটি গ্লোবাল ভেরিয়েবল, যার মধ্যে সংরক্ষিত আছে মোট রেকর্ড সংখ্যা- এটা ইনডেক্স সংখ্যা থেকে ১ বেশি। রেকর্ড থাকলে আমরা পরের লাইনে যেয়ে প্রথমে cb নামক একটি কমান্ড বিল্ডার অবজেক্ট ডিক্লায়ার করেছি। এর প্যারামিটার হলো ডাটাএডাপটার (da)। পরের লাইনে আমাদের ডাটাসেট [ds] এর মাধ্যমে পয়েন্ট করছি ঠিকানা ডাটাবেইজের টেবিল থেকে একটি তথ্যলাইন মুছে ফেলতে। Rows(RecNum) নির্দেশ করছে কোন্‌টি মুছা হবে। মনে রাখা দরকার, বর্তমানে প্রদর্শিত রেকর্ডের ইনডেক্স সংখ্যা হলো RecNum। সুতরাং আমরা এটিই ডিলিট করার কথা বলেছি। ঠিকানা ডাটাবেইজ এরপর আপডেট করতে হবে। বাস্তবে এই আপডেট নির্দেশই এক্সেসের ফিজিক্যাল ফাইলে রবদল নিশ্চিত করে। পরের লাইনের কাজ হলো ইনডেক্স সংখ্যা ১ কমানো- কারণ এখন একটি ইনডেক্স কমে গেছে। এরপর একটি মেসেজ বাক্সে রেকর্ড ডিলিটের বার্তাটি ব্যবহারকারীকে দেওয়া হয়েছে। আর সবশেষে কানেকশন বন্ধ করার নির্দেশ এসেছে।

এবার ‘এই রেকর্ড এডিট করুন’ ও ‘সেইভ করুন’ বোতামদ্বয়ের কোড নিয়ে আলোচনার প্রয়াস পাচ্ছি। আর এরই মাধ্যমে ডাটাবেইজ প্রোগ্রামিংয়ের ওপর আলোচিত এ পরিচ্ছেদের ইতি টানবো। শিক্ষার্থীরা এ থেকে যা শিখেছেন, সাধারণ ডাটাবেইজ প্রোগ্রামিংরে জন্য যথেষ্ট হবে বলে মনে করছি। তবে এর ওপর আরো গভীর জ্ঞানার্জনের জন্য ভিজুয়্যাল বেসিক.নেট দিয়ে ডাটাবেইজ সৃষ্টির উন্নতমানের বই-পুস্তক পাঠ ও ইন্টারনেটে প্রাপ্ত তথ্যাদি অধ্যয়ন করতে পারেন।

```
'editing btn code
Private Sub btnEdit_Click(sender As Object, e As EventArgs) Handles btnEdit.Click
    ' confimation test
    Dim ResP As Integer = MsgBox("এই রেকর্ডে রদবদল করবেন?", MsgBoxStyle.YesNo)
    If ResP = 7 Then
        Exit Sub
    End If
    EnableFourFlds() 'enable fields for editing
    btnUpdate.Visible = True 'show the update button
    Button1.Enabled = False
    btnDelete.Enabled = False
    btnNewRecord.Enabled = False
End Sub
```

আমাদেরকে আরোও একটি টেস্ট করতে হবে। কারণ, ব্যবহারকারী এই বোতামে ভুলেও ক্লিক করতে পারেন। টেস্টে পাশ হওয়ার পর প্রথমে ফিল্ডগুলো সক্রিয় করতে একটি সাবরুটিন ডাক দিয়েছি। এরপর সেইভ করুন বোতামটি দৃশ্যমান করেছি। এছাড়া ‘নতুন রেকর্ডের প্রস্তুতি নিন’, ‘ডাটা আনুন’ এবং ‘এই রেকর্ড ডিলিট করুন’ -এ তিনটি বোতামকে নিষ্ক্রিয় করতে হয়েছে। এগুলোর কোনো ফাংশন নেই এডিটের পূর্ব পর্যন্ত।

১৭তম পরিচ্ছেদ: *Database Programming Made Simple*

সহজ ডাটাবেইজ প্রোগ্রামিং: *Database Programming Made Simple*

```
Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles btnUpdate.Click
    Dim ResP As Integer = MsgBox("আপনি কি সব রেকর্ড সঠিকভাবে প্রবেশ করেছেন?", MsgBoxStyle.YesNo)
    If ResP = 7 Then
        Exit Sub
    End If
    UpdateRecord() 'call update sub
    MakeFourFldReadOnly() 'call sub to lock filds
    GetAndFillRecord() '□□  □□" #□□□$ □□%□□□□ &□□ □□
    btnUpdate.Visible = False
    Button1.Enabled = True
    btnDelete.Enabled = True
    btnNewRecord.Enabled = True
End Sub
```

এখানেও একটি টেস্ট আছে! ব্যবহারকারী সঠিকভাবে সবকিছু এডিট করেছেন কি না জেনে নিতে হবে। টেস্টের ফলাফল ইতিবাচক হলে আমরা কল করতে পারি UpdateRecord() নামক সর্বশেষ সাবরুটিনটি। আপডেট হওয়ার পর চারটি ফিল্ড রিড-অনলি বানানো ও একটি বোতাম নিষ্ক্রিয় ও তিনটি সক্রিয় করা দরকার। শেষের একটি সাবরুটিন ও চারটি লাইন এ কাজগুলো সারবে।

এবার দেখে নেই সর্বশেষ কোড ব্লকটি। এটা একটি সাবরুটিন যার কাজ হলো এডিট করা লেখাগুলো স্ব স্ব স্থানে নিয়ে যাওয়া- তথা তথ্যাদি আপডেট করা।

```
Private Sub UpdateRecord()
    Dim cb As New OleDb.OleDbCommandBuilder(da) 'commandBuilder Object creation
    ds.Tables("TikanaDB").Rows(RecNum).Item(1) = NameText.Text
    ds.Tables("TikanaDB").Rows(RecNum).Item(2) = AddressText1.Text
    ds.Tables("TikanaDB").Rows(RecNum).Item(3) = AddressText2.Text
    ds.Tables("TikanaDB").Rows(RecNum).Item(4) = MobileText.Text
    da.Update(ds, "TikanaDB") 'update database
    MsgBox("রেকর্ড রদবদল হয়েছে।") 'show success
    con.Close() 'close connection
End Sub
```

সেইভ করুন [আপডেট করার] বোতামটির কোড বাস্তবে খুব এটা কঠিন কিছু না। প্রথমে -এর নতুন একটি ইন্সট্যান্ট OleDb.OleDbCommandBuilder সৃষ্টি করতে হবে। এর প্যারামিটার হবে আমাদের সৃষ্ট ডাটাএডাপটার [da]। এরপর চারটি লাইনে ডাটাসেটে নিয়ে যেতে হবে নির্দিষ্ট ইনডেক্সের [ds] চারটি আইটেম [১,২,৩ ও ৪]। সব প্রস্তুত। এবার da.Update(ds, "TikanaDB") লাইনটির মাধ্যমে এক্সেস ডাটাবেইজের টেবিলটি আপডেট করা হয়েছে। এরপর মেসেজ বাক্সের মাধ্যমে সফলতা জানানো ও কানেকশন ক্লোজ করা হয়েছে। আলহামদুলিল্লাহ! ঠিকানা ডাটাবেইজের কাজ পূর্ণাঙ্গভাবে শেষ হলো। শিক্ষার্থীদের প্রতি আবারো অনুরোধ, শুরু থেকে শেষ পর্যন্ত কোডের ব্যাখ্যা প্রয়োজনে একাধিকবার পাঠ করে বুঝার চেষ্টা করবেন। আর অবশ্যই, নিজে এই ডাটাবেইজ প্রোগ্রামটি সৃষ্টির প্রাকটিস করবেন।

শেষ হলো ডাটাবেইজ প্রেগ্রামিংয়ের ওপর রচিত এ পরিচ্ছেদটি। এবার আরো এক দু'টো স্টেপ বাই স্টেপ মডেল প্রোগ্রাম বানানোর প্রশিক্ষণ উপস্থাপনের চেষ্টা করবো। আসুন, আপনাদের সহযোগিতায় গড়ে তুলি ওসব ব্যবহারেপযোগী এ্যাপলিকেশন।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

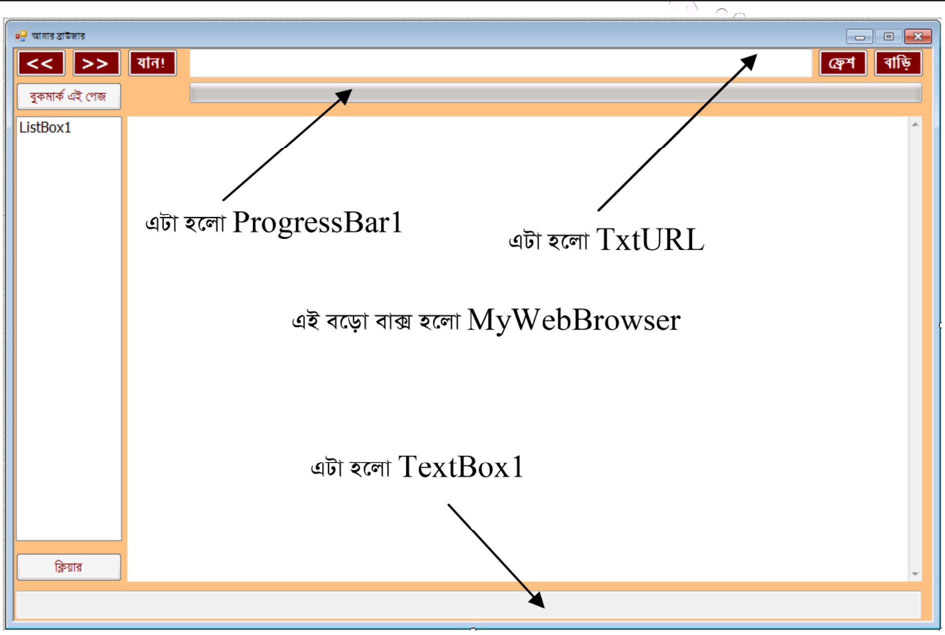
এ পরিচ্ছেদে আরো কয়েকটি নমুনা এ্যাপলিকেশন পদে পদে আমরা গড়ে তুলবো। শিক্ষার্থীরা প্রতিটি পদক্ষেপ সঠিকভাবে অনুসরণ করবেন, যেভাবে ইতোমধ্যে এ গ্রন্থে আলোচিত সকল প্রোগ্রাম তৈরির সময় করেছেন। পরিচ্ছেদ ১৭ শেষ হওয়ার পর ভেবেছিলাম ভিজুয়্যাল বেসিক.নেট এর উপর রচিত এই বইটির ইতি টানবো। কিন্তু শেষ পর্যন্ত সিদ্ধান্ত পরিবর্তন করলাম এ কারণে যে, কমিম্পউটার সফটওয়্যার সৃষ্টির মূলে প্রাকটিস একটি গুরুত্বপূর্ণ ব্যাপার। শিক্ষার্থীরা যতো বেশি প্রাকটিস করবেন ততো বেশি প্রোগ্রামিংয়ের বিভিন্ন কনসেপ্ট সম্পর্কে অভিজ্ঞতা অর্জন করবেন। হতে পারে, এ গ্রন্থের পাঠকদের মধ্যে ভবিষ্যতে দেশের একজন উজ্জ্বল তারকায়ও কেউ রূপান্তরিত হবেন। আমি অবশ্যই এ আশা করবো। যাক, আমরা একে একে এই নমুনা এ্যাপলিকেশনগুলোর ওপর বিস্তারিত আলোচনা শুরু করছি, এখনই।

১. একটি ব্যক্তিগত ওয়েভ ব্রাউজার: A Personal Web Browser

হ্যাঁ, ভিজুয়্যাল বেসিক.নেট ও ভিজুয়্যাল স্টুডিও আপনাকে ওয়েভ ব্রাউজার এ্যাপলিকেশনের মতো জটিল প্রোগ্রাম সৃষ্টির সুযোগ তৈরি করে দিয়েছে। আর তা এতোই সহজ! একটু পরই শিক্ষার্থীরা বুঝতে পারবেন। কয়েকটি বিশষ পদক্ষেপ ও অতি অল্প কিছু কোডিং দ্বারাই তৈরি হবে আপনার নিজস্ব স্বাপ্নিক ওয়েভ ব্রাউজার এ্যাপলিকেশন।

ক. ডিজাইন: নিচে একটি চিত্র দিয়েছি। আপনার ওয়েভ ব্রাউজার এরূপই হবে। এতে আছে ' টি বোতাম, ২টি টেক্সট বাক্স, ১টি লিস্ট বাক্স, ১টি প্রগ্রেস বার ও ১টি ওয়েভ ব্রাউজার কন্ট্রোল। এগুলোর অবস্থান এ্যারো দ্বারা ছবিতে দেখিয়েছি। এদের প্রত্যেকটির বিভিন্ন প্রোপার্টি নিচের টেবিলে তুলে ধরেছি। টুল বাক্স থেকে একে একে এগুলো ফর্মে নিয়ে আসুন। এরপর নির্দেশ মুতাবিক এদের প্রোপার্টি সেট করুন।

নাম	টেক্সট প্রোপার্টি	নাম	টেক্সট প্রোপার্টি	নাম	টেক্সট প্রোপার্টি
BtnBack	<<	BtnAddToFav	বুকমার্ক এই পেইজ	ListBox1	Font: Tahoma, 12pt
BtnForward	>>	BtnClear	ক্লিয়ার	ProgressBar1	--
BtnGo	যান!	TxtURL	--	MyWebBrowser	--
BtnRefresh	ফ্রেশ	TextBox1	ReadOnly: True	Form1	আমার ব্রাউজার
BtnHome	বাড়ি				



১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

২. কোডিং: আমরা ডিজাইন করে নিয়েছি। আপনি চাইলে ডিজাইনটি অন্যরকমও করতে পারেন। তবে বোতাম, টেক্সবাক্স, লিস্টবাক্স ইত্যাদি তো অবশ্যই থাকতে হবে। নিচে সম্পূর্ণ কোড তুলে ধরেছি। পরে ব্যাখ্যা আসবে।

```
Public Class Form1
   'যান! বোমাতের কোড। TxtURL -এতে পূর্ণাঙ্গ ওয়েভসাইট ঠিকানা লিখা থাকবে, তার ওপর অনুসন্ধান চালাবে এই বোতামটি।
    Private Sub BtnGO_Click(sender As Object, e As EventArgs) Handles BtnGo.Click
        MyWebBrowser.Navigate(TxtURL.Text) 'ন্যাভিগেট হলো সিস্টেমের ফাংশন। এর একটি প্যারামিটার হচ্ছে টেক্সট
        MyWebBrowser.Focus() 'ব্রাউজার বাক্সে ক্লিক করার অনুরূপ
    End Sub
   'এটা বাড়ি বোতামের কোড। আমরা গুগলস্ সার্চকে হোম পেইজ বানিয়েছি।
    Private Sub BtnHome_Click(sender As Object, e As EventArgs) Handles BtnHome.Click
        MyWebBrowser.Url = New System.Uri("https://www.google.com/webhp?hl=en&tab=wi")
        MyWebBrowser.Focus()
    End Sub
   'এটা আগের ওয়েভ পেইজে ফিরে যাওয়ার বোতাম।
    Private Sub BtnBack_Click(sender As Object, e As EventArgs) Handles BtnBack.Click
        MyWebBrowser.GoBack()
        MyWebBrowser.Focus()
    End Sub
   'এটা পরের ওয়েভ পেইজে যাওয়ার বোতাম।
    Private Sub BtnForward_Click(sender As Object, e As EventArgs) Handles BtnForward.Click
        MyWebBrowser.GoForward()
        MyWebBrowser.Focus()
        TextBox1.Text = MyWebBrowser.Url.ToString
    End Sub
   'এটা বর্তমান ওয়েভ পেইজ রেফ্রেশ করার বোতাম।
    Private Sub BtnRefresh_Click(sender As Object, e As EventArgs) Handles BtnRefresh.Click
        MyWebBrowser.Refresh()
        MyWebBrowser.Focus()
    End Sub
   'পেছনের ওয়েভ পেইজে যাওয়া শেষ হলে এই ইভেন্ট বোতামকে নিষ্ক্রিয় করবে।
    Private Sub MyWebBrowser_CanGoBackChanged(sender As Object, e As EventArgs) Handles
MyWebBrowser.CanGoBackChanged
        BtnBack.Enabled = MyWebBrowser.CanGoBack
    End Sub
   'সামনের ওয়েভ পেইজে যাওয়া শেষ হলে এই ইভেন্ট বোতামকে নিষ্ক্রিয় করবে।
    Private Sub MyWebBrowser_CanGoForwardChanged(sender As Object, e As EventArgs) Handles
MyWebBrowser.CanGoForwardChanged
        BtnForward.Enabled = MyWebBrowser.CanGoForward
    End Sub
   'ন্যাভিগেট হওয়ার পর এই ইভেন্ট সৃষ্টি হয়। এটে ধরে আমরা ইউআরএলকে উপরের ট্যাক্সট বাক্সে নিয়ে গেছি।
    Private Sub MyWebBrowser_Navigated(sender As Object, e As WebBrowserNavigatedEventArgs) Handles
MyWebBrowser.Navigated
        TxtURL.Text = MyWebBrowser.Url.ToString
    End Sub
   'ওয়েভ পেইজ লোড হওয়ার সময় প্রগ্রেস বারের ক্রিয়ার বোতাম।
    Private Sub MyWebBrowser_Navigating(sender As Object, e As WebBrowserNavigatingEventArgs) Handles
MyWebBrowser.Navigating
        With ProgressBar1 'উইথ--এন্ড উইথ ব্লক দ্বারা এটা বাড়ানোর রেইট নির্ধারণ করা হয়েছে।
            .Minimum = 0
            .Maximum = 52
```

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
            .Step = 4
        End With
            For index As Integer = 0 To 52 Step 4
                ProgressBar1.Value = index
                System.Threading.Thread.Sleep(35) 'নির্দিষ্ট সংখ্যক [৩৫] মিলিসেকেন্ডের জন্য বন্ধ থাকা
            Next
    End Sub
    'ফর্ম লোডের সময় ডিফোল্ট ওয়েব পেইজ খোলার নির্দেশ। আমরা গুগলসের সার্চ পেইজ নির্দিষ্ট করেছি।
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
      MyWebBrowser.Navigate("https://www.google.com/webhp?hl=en&tab=wi")
        For Each item As String In My.Settings.Bookmarks
            ListBox1.Items.Add(item) 'বুকমার্কের সেটিংয়ে সংরক্ষিত ওয়েভপেইজ ঠিকানা লিস্ট বাক্সে নিয়ে আসা।
        Next
    End Sub
    'এটা বুকমার্ক ক্লিয়ার বোতামের কোড।
    Private Sub BtnClear_Click(sender As Object, e As EventArgs) Handles BtnClear.Click
        Dim ResP As Integer = MsgBox("সব বুকমার্ক মুছে দেবেন?", MsgBoxStyle.YesNo)
        If ResP = 7 Then 'no
            Exit Sub 'না মুছেই
        End If
        My.Settings.Bookmarks.Clear()
            ListBox1.Items.Clear()
        End Sub
    'এটা বর্তমান পেইজ বুকমার্ক বোতামের কোড।
    Private Sub BtnAddToFav_Click(sender As Object, e As EventArgs) Handles BtnAddToFav.Click
        My.Settings.Bookmarks.Add(MyWebBrowser.Url.ToString) 'সেটিংয়ে পেজ ঠিকানা যাবে।
        ListBox1.Items.Clear() 'লিস্ট ক্লিয়ার হবে।
     For Each item As String In My.Settings.Bookmarks
            ListBox1.Items.Add(item) 'সেটিংয়ের সকল পেজ ঠিকানা একে একে লিস্ট বাক্সে আসবে।
        Next
    End Sub
    'লিস্ট বাক্সে ডবল ক্লিক করে ওয়েভ পেইজে যাওয়ার কোড।
    Private Sub ListBox1_DoubleClick(sender As Object, e As EventArgs) Handles ListBox1.DoubleClick
        MyWebBrowser.Navigate(ListBox1.SelectedItem) 'লিস্ট বাক্সের আইটেম প্যারামিটার।
    End Sub
    'প্রাপ্ত ওয়েভ পেইজের ঠিকানা প্রাপ্তি ও এর ইউআরএল প্রদর্শন।
   Private Sub MyWebBrowser_StatusTextChanged(sender As Object, e As EventArgs) Handles
MyWebBrowser.StatusTextChanged
         TextBox1.Text = MyWebBrowser.StatusText 'প্রাপ্ত ওয়েভ পেইজের ঠিকানা নিচের টেক্সবাক্সে যাবে।
      End Sub
End Class
```

ব্রাউজারের মতো জটিল একটি এ্যাপলিকেশনের কোড এটুকুই মাত্র! আমরা মাইক্রসফটকে ধন্যবাদ না দিয়ে পারবো না। তারা .নেট ফ্রেমওয়ার্ক টেকনোলজি ডেভেলাপ করে প্রোগ্রামারদের জন্য বিরাট সম্ভাবনার দরোজা খুলে দিয়েছেন। ভিজুয়্যাল বেসিক.নেট ফ্রেমওয়ার্কের কিছু গুরুত্বপূর্ণ ক্লাস লাইব্রেবির যেগুলো টুলবাক্সের সঙ্গে সম্পৃক্ত, তাদের একটি লিস্ট গ্রন্থের পরিশিষ্টে উদাহরণ ব্যবহারসহ যুক্ত করার আশা করছি। যাক, এবার আমাদের ব্যক্তিগত ওয়েভ ব্রাউজারের কোডগুলোর ওপর প্রয়োজনীয় কিছু ব্যাখ্যা তুলে ধরছি।

প্রথমত জরুরী সবগুলো কোডের ব্যাখ্যা কমেন্ট দ্বারা করা হয়েছে। দ্বিতীয়ত কোড লাইনের অধিকাংশই স্ব-ব্যাখ্যাত- অর্থাৎ কোডই বুঝাচ্ছে তার ক্রিয়া কি হবে। তবে শিক্ষার্থীদের সুবিধার্তে ফর্মের ক্লাসের ভেতরস্থ ১৪টি সাবরুটিন সম্পর্কেই আরো কিছু আলোচনা করছি।

১. Private Sub BtnGO_Click -এটা হলো 'যান!' বোতামের ক্লিক সাবরুটিন। আপনি উপরের টেক্সবাক্সে যে ওয়েভ

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

ঠিকানা লিখবেন তা সঠিক হলে সেই ওয়েভ সাইট খুলবে।

২. Private Sub BtnHome_Click -এটা হলো 'বাড়ি' বোতামের ক্লিক সাবরুটিন। আমরা গুগল্স সার্চ পেইজকে বাড়ি বা হোম পেইজ বানিয়েছি।

৩. Private Sub BtnBack_Click -এটা হলো '<<' বোতামের ক্লিক সাবরুটিন। আগের পৃষ্ঠা [যদি থেকে থাকে], এই বোতাম আবার খুলবে।

৪. Private Sub BtnForward_Click -এটা হলো '>>' বোতামের ক্লিক সাবরুটিন। পরের পৃষ্ঠা [যদি থেকে থাকে], এই বোতাম আবার খুলবে।

৫. Private Sub BtnRefresh_Click -এটা হলো 'ফ্রেশ' বোতামের ক্লিক সাবরুটিন। বর্তমান পেইজটি পুনরায় লোড করবে এই ক্লিক ইভেন্টটি।

৬. Sub MyWebBrowser_CanGoBackChanged -এটা হলো 'GoBack' ফাংশনের একটি ইভেন্ট। এটা আত্মপ্রকাশ করে পেছনের পৃষ্ঠায় যেতে পারলে। আমরা এই ইভেন্টকে কাজে লাগিয়ে বলেছি আগের পৃষ্ঠায় যাওয়ার বোতাম সক্রিয় হতে পারে।

৭. Sub MyWebBrowser_CanGoForwardChanged -এটা হলো 'GoForward' ফাংশনের একটি ইভেন্ট। আমরা এ ইভেন্টকে কাজে লাগিয়ে বলেছি পরে যাওয়ার বোতামটি সক্রিয় হতে পারে।

৮. Sub MyWebBrowser_Navigated -এটা হলো 'Navigation' ফাংশনের একটি ইভেন্ট। এই ইভেন্ট সৃষ্টি হয় ওয়েভ পেইজ লোড হয়ে শেষ হওয়ার পরমুহূর্তে। আমরা একে ধরে সত্যিকার ইউআরএলকে স্ট্রিং টাইপে পরিবর্তন করে উপরের টেক্সট্ বাক্সে দেখিয়েছি।

৯. Sub MyWebBrowser_Navigating -এটাও ওয়েভ ব্রাউজারের একটি ইভেন্ট। পেইজ লোড হওয়ার সময় এটি সক্রিয় হয়। আমরা একে ধরে লোড হওয়ার দৃশ্যত কাজ প্রদর্শনের ব্যবস্থা করেছি প্রগ্রেসবার কন্ট্রোল ব্যবহার করে। লক্ষ করুন, ০ থেকে ১০০ পর্যন্ত ৪ করে অগ্রসর হওয়ার রেট নির্ধারিত করেছি প্রথমেই ইউথ-এন্ড উইথ ব্লকে। অর্থাৎ তার পুরোটাই ২৫টি পদক্ষেপে বাম দিক থেকে ডান দিকে যাবে। রেট নির্ধারিত হওয়ার পর একটি ফর -- নেক্সট্ লুপ দ্বারা সত্যিকার প্রগ্রেস হওয়ার গ্রাফিক্যাল এ্যানিমেশন দেখানো হয়েছে। আমরা .নেট এর এপিআই - System.Threading.Thread.Sleep(35) কাজে লাগিয়েছি। কতটুকু সময় প্রতিটি স্টেপে থামবে তা নির্ধারিত হয়েছে এই এপিআই থেকে। এখানে ৩৫ মিলিসেকেন্ড বুঝাচ্ছে [১০০০ মি.সে. = ১ সেকেন্ড]।

১০. Sub Form1_Load -এটা হলো ফর্ম লোড হওয়ার সময়কার ইভেন্ট। আমরা চাই প্রথমেই গুগলস্ এর সার্চ ওয়েভ পেইজটি আসবে। সুতরাং ফর্ম খোলার পরই এই পেইজটিতে ন্যাভিগেট করার জন্য নির্দেশ দিয়েছি প্রথম লাইনে। এরপর আমরা এটাও চাই যে লিস্ট বাক্সে সেটিংয়ে সংরক্ষিত বুকমার্কসগুলো লাইনে লাইনে প্রদর্শন করতে। ফর ইচ্ -- নেক্সট্ লুপের কাজ এটি। শিক্ষার্থীদের একটি বিষয় জানা থাকা দরকার- কিভাবে সেটিংয়ে যেয়ে বুকমার্কস্ তৈরি করা যায়। এ ব্যাপারে পরের পৃষ্ঠার চিত্রগুলো দেখুন ও নিচের ব্যাখ্যাগুলো পাঠ করুন।

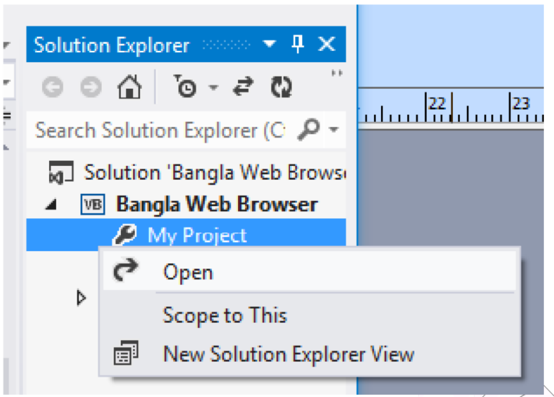
প্রথমে সলুশন এক্সপ্লোরারে যেয়ে মাই প্রজেক্টে রাইট-ক্লিক করুন। ফলে মাঝের জানালা আসবে। সেখানে সেটিং ক্লিক করুন। প্রথম কলামে উপরেই লিখুন Bookmark । এবার ডানের [...] বোতাকে ক্লিক করলেই একটি জানালা আসবে। সেখানে এই স্ট্রিংটি লিখুন: https://www.google.com/imghp?hl=en&tab=wi । এরপর OK ক্লিক করুন। ব্যস! সেটিংয়ে একটি স্ট্রিং সংরক্ষিত হবে। ভবিষ্যতে আপনি ব্রাউজারে প্রদর্শিত যে কোনো পেইজ সংরক্ষণের জন্য শুধুমাত্র **'বুকমার্ক এই পেইজ'** বোতামে ক্লিক করলেই সেটিংয়ের লিস্টে ঐ পেইজের ইউআরএল তালিকাভুক্ত হবে। আপনি যতোটা ইচ্ছা পেইজের ইউআরএল বুকমার্ক করতে পারেন। আর লিস্টের যে কোনো লাইনে ডবল ক্লিক করলেই সেই ইউআরএল-এ ন্যাভিগেট করবে ওয়েভ ব্রাউজার।

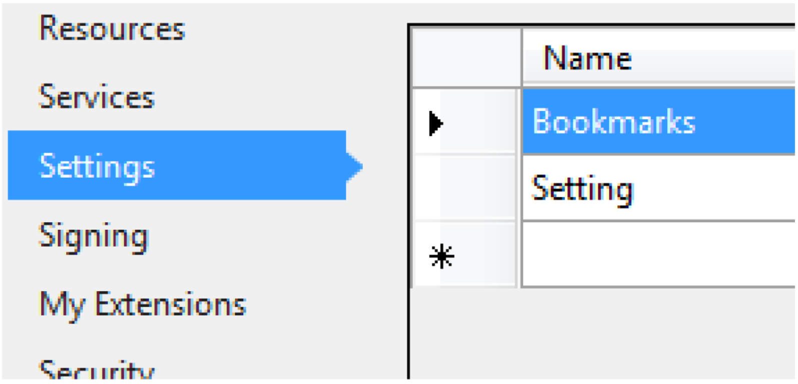
১১. Sub BtnClear_Click -এটা হলো 'ক্লিয়ার' বোতামের ক্লিক সাবরুটিন। বুকমার্ককৃত সকল পেইজের ইউআরএল এই বোতামে ক্লিক করে মুছে দিতে পারেন। অবশ্য একটি ওয়ার্নিং থাকবে- যা আমরা মেসেজ বাক্স ফাংশন দ্বারা প্রদর্শনের ব্যবস্থা করেছি।

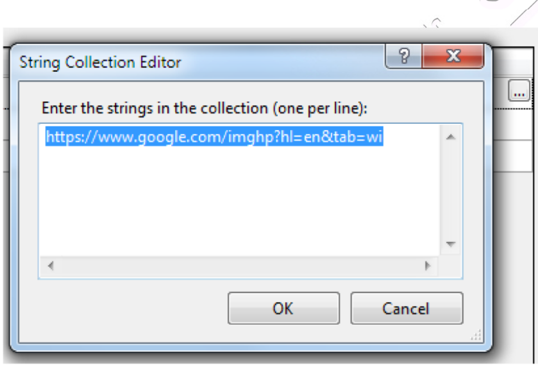
১২. Sub BtnAddToFav_Click -এটা হলো **'বুকমার্ক এই পেইজ'** বোতামের ক্লিক সাবরুটিন। প্রদর্শিত ইউআরএল সেটিংয়ের বুকমার্কস্ লিস্টে যাবে। তবে লিস্ট বাক্সেও নতুন বুকমার্ক আনতে হবে। এজন্য প্রথমে আমরা লিস্ট বাক্সের সকল আইটেম মুছে দিয়েছি দ্বিতীয় লাইনের নির্দেশ দ্বারা। এরপর একটি ফর ইচ-নেক্সট্ লুপ কাজে লাগিয়ে সেটিং থেকে পুনরায় সকল আইটেম নিয়ে

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*







এসেছি শূন্য লিস্টবাক্সে।

১৩. Sub ListBox1_DoubleClick -এটা হলো লিস্ট বাক্সের ডবলক্লিক ইভেন্ট। একে ধরে আমরা সিলেক্ট করা লাইন মুতাবিক ইউআরএল-এ ন্যাভিগেশন করার জন্য নির্দেশ করেছি ওয়েভব্রাউজারকে।

১৪. MyWebBrowser_StatusTextChanged -এটা হলো ওয়েভব্রাউজারের স্টেটাসটেক্সটচেঞ্জড নামক ইভেন্ট সাবরুটিন। আমরা একে ধরে প্রদর্শিত ওয়েভ পেইজের ইউআরএল লোড হওয়ার স্টেটাস টেক্সটটুকু নিচের টেক্সট বাক্সে প্রদর্নের জন্য নিয়ে এসেছি। লোড সম্পন্ন হয়ে গেলে, 'Done' শব্দটি বাক্সে দেখাবে। এবার দেখে নিন আপনার ব্রাউজারের একশন।



১৬তম পরিচ্ছেদ: *Few More Sample Applications*

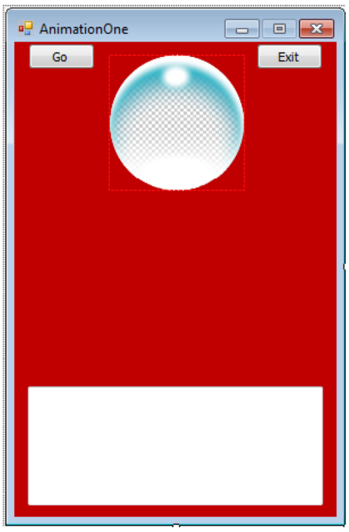
আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

২. সাধারণ এ্যানিমেশন: Simple Animation

এই গ্রন্থে আমরা বেশ কয়েকটি উদাহরণ প্রোগ্রাম নিয়ে বিস্তারিত আলেচনা করেছি। ভিজুয়্যাল বেসিক.নেট যে একটি শকিক্তশালী সফ্‌টওয়্যার ডেভেলাপমেন্ট ভাষা তা- তো নিশ্চয়ই পাঠকরা এখন অনুধাবন করে নিয়েছেন। এখন আমরা আরেকটি ছোট্ট উদাহরণ তুলে ধরছি। এটা হলো এ্যানিমেশন কোডিং উদাহরণ।

উদ্দেশ্য: ১. একটি বলকে উপর থেকে নিচে ধীরে ধীরে নামিয়ে আনবো। ২. একই সময় স্বয়ংক্রিয়ভাবে কিছু বাংলা লেখা আসবে একটি টেক্সট্ বাক্সে।

ক. ইন্টারফেইস ডিজাইন: একটি নতুন ইউন্ডোজ ফর্ম এ্যাপলিকেশন তৈরি করুন। এর নামকরণ যা ইচ্ছে করতে পারেন। আমি নামকরণ করেছি: Animation1। ফর্মের সাইজ হবে: ৩০০,৪৫৭। ফর্মের FormBorderStyle হবে - FixedSingle । ফর্মের মধ্যে থাকবে একটি টেক্সট্ বাক্স, দু'টি বোতাম ও একটি পিকচারবক্স। পিকচারবাক্সের BackgroundImage হবে যে কোনো একটি .png ছবি। ইন্টারনেটে প্রচুর ছবি মিলে। প্রয়োজনে ফটোশপে এনে পিএনজি হিসাবে বানাতেও পারেন। আমি একটি বল ব্যবহার করেছি। যেখানে ছবির ফাইল থাকবে সেখানে ব্রাউজ করে রিসোর্সে নিয়ে আসবেন। ছবির নাম যা থাকবে তা অবশ্য কোডে ব্যবহৃত হবে। ৩টি টাইমার কন্ট্রোল। এদের নামকরণ করুন: aTimer, wTimer, tTimer । টাইমারগুলো অবশ্যই ফর্মের উপরে থাকবে না। টাইমার কন্ট্রোলে ডবল ক্লিক করলেই এগুলো এসে যাবে। এরপর প্রতিটির প্রোপার্টি দেখে নামকরণ বদলাবেন। নিচের চিত্রে আমাদের ইন্টারফেইস ডিজাইনটি তুলে ধরেছি- একটু দেখে নিন।



খ. কোডিং: এ্যানিমেশন কোডিং কিছুটা জটিল- কিন্তু তেমন কঠিন নয়! প্রথমেই ফর্মে ডবল ক্লিক করে কোড জানালা খুলুন। সাবরুটিনগুলো শুরুর আগে দেখানো স্থানে ৫টি গ্লোবাল ডিক্লারেশন দরকার:

```
Public Class Form1
    Dim waitTime As Integer = 500 'millisecond to control speed of movement
    Dim speed As Integer = 60 'ms actual speed of movement
    Dim OXpos As Integer : Dim OYPos As Integer 'to keep record of the PicuteBox position in the
form
    Dim speedText As Integer = 500 'ms to cotrol typing speed
    'text to be typed. Unicode is typed with Bijoy Bayanno Unicode option.
    Dim printText As String = "ভিজুয়্যাল বেসিক.নেট -নতুন শিক্ষার্থীদের জন্য গুরুত্বপূর্ণ প্রশিক্ষণ রিসোর্স -
গ্রন্থকার ইঞ্জিনিয়ার আজিজুল বারী, ডিপ্লোমা ইন সাইন্স এন্ড টেকনোলজি, ইংল্যান্ড।"
```

এগুলো আমরা পরে ব্যবহার করবো। এরপর নিচের সাবরুটিনটি যাবে। এটা হলো aTimer -এর Tick নামক ইভেন্ট সাবরুটিন।

```
'this sub animates picture and types text
Private Sub aTimer_Tick(sender As Object, e As EventArgs) Handles aTimer.Tick
    TextBox1.Text += printText.Substring(TextBox1.Text.Length, 1) 'one by one printText writing will be printed in the text box
    If TextBox1.Text = printText Then aTimer.Enabled = False 'Once writing is finished, the timer action will stop
    'same time picture box's location will change by y+1 (1 point down). Notice x value doesn't change.
    MyPictureBox.Location = New Point(MyPictureBox.Location.X, MyPictureBox.Location.Y + 1)
    If MyPictureBox.Location.Y = 205 Then '250 point (y) is the minimum.
      aTimer.Enabled = False 'at that point timer stops
    End If
End Sub
```

উপরের সাবটিই মূল এ্যানিমেশনের কাজটি করে। পরের সাবরুটিন দু'টি টাইমার কন্ট্রোল সক্রিয় ও একটি অক্রিয় করে।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
'this sub enales aTimer and tTimer and disables wTimer
   Private Sub wTimer_Tick(sender As Object, e As EventArgs) Handles wTimer.Tick
      wTimer.Enabled = False
      aTimer.Enabled = True
      tTimer.Enabled = True
   End Sub
```

এর পরেরটি হলো ফর্ম লোড ইভেন্ট সাবরুটিন। আমরা একে ধরে প্রথমে ডিক্লায়ার করা ভেরিয়েবলগুলোতে ভ্যালু প্রবেশ করেছি।

```
'form load-time inirialization of 3 variables associated with three timers
   Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
      aTimer.Interval = speed
      wTimer.Interval = waitTime
      tTimer.Interval = speedText
      ' keep a record of PictureBox Position
      OXpos = MyPictureBox.Location.X : OYPos = MyPictureBox.Location.Y
   End Sub
```

নিচের ছোট্ট সাবটি হলো Exit বোতামের ক্লিক ইভেন্ট কোড। বুঝতেই পারছেন, এটি দ্বারা প্রোগ্রাম বন্ধ করা হয়।

```
 'exit- here Me means the Form itself.
   Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      Me.Close()
   End Sub
```

ব্যস! কোড মাত্র এটুকুই। এবার টেস্ট রানের পালা। নিচের তিনটি চিত্রে এ্যানিমেশনকালীন স্ক্রিনশট তুলে ধরেছি।



৩. বাংলা-ইংরেজি-বাংলা তারিখ রূপান্তর: Bangla-English-Bangla Date Conversion

ইনটারনেটে একাধিক ওয়েভসাইট আছে যেখানে বাংলা থেকে ইংরেজি এবং ইংরেজি থেকে বাংলা তারিখ রূপান্তরের সুযোগ আছে। কিন্তু কিভাবে এরূপ এ্যাপলিকেশন ডেভেলাপ করা যায়? এ প্রশ্নের জবাব খুঁজতে যেয়েই শেষ পর্যন্ত নিজেই নতুন এই প্রোগ্রাম সৃষ্টি করতে সক্ষম হয়েছি। আর এটি শেয়ার করতে চাই সকল শিক্ষার্থীদের সাথে। এই প্রোগ্রাম ডেভেলাপমেন্ট টেকনিক থেকে ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং ভাষা সম্পর্কেও অনেক ব্যাপার জানা ও সেসাথে অভিজ্ঞতা অর্জিত হবে। তাহলে আসুন, দেখে নিই কিভাবে এ ব্যবহারযোগ্য কার্যকরী এ্যাপলিকেশনটি বানাতে হয়।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

তথ্যাদি সংগ্রহ: প্রথমেই জানতে হবে বাংলা ও ইংরেজি তারিখের মধ্যে সম্পর্ক কি। কিছু অনুসন্ধানের পর জানা গেলো, বাংলা সন বের করতে ৫৯৩ বিয়োগ দিতে হবে ইংরেজি সন থেকে। তবে তারিখ যদি ১লা জানুয়ারি থেকে ১৩ এপ্রিলের মধ্যে হয় তাহলে কিন্তু বিয়োগ দিতে হবে ৫৯৪। এর কারণ হলো তখনো বাংলা সন বদলাবে না। এর পরের সকল তারিখের ক্ষেত্রে অবশ্য বিয়োগ দিতে হবে ৫৯৩। তাহলে সন পরিবর্তনের ফর্মুলা হলো:

১লা জানুয়ারি থেকে ১৩ এপ্রিল পর্যন্ত তারিখের ভেতর: ইংরেজি সন - ৫৯৪ = বাংলা সন। [বাংলা সন = +৫৯৪]
১৪ এপ্রিল থেকে ৩১ ডিসেম্বর পর্যন্ত তারিখের ভেতর: ইংরেজি সন - ৫৯৩ = বাংলা সন। [বাংলা সন = +৫৯৩]

তারিখ পরিবর্তের ক্ষেত্রে একটি ফর্মুলা জানা গেছে। তাহলো: **১৪ ১৫১৫ ১৬১৬১৬১৬ ১৫১৫ ১৪ ১৩ ১৫**। মনে রাখার জন্য আমরা লিখতে **৪ ৫৫ ৬৬৬৬ ৫৫ ৪৩৫**। এখানে ১লা বৈশাখ শুরু হয় ইংরেজি এপ্রিল মাসের ১৪ তারিখ। ১লা জৈষ্ঠ্য হলো ইংরেজি মে মাসের ১৫, ১লা আষাঢ় শুরু হয় ইংরেজি জুন মাসের ১৫ ইত্যাদি বুঝাচ্ছে। এই ফর্মুলা থেকে আমরা একটি তথ্যের টেবিল তৈরি করতে পারি। যেমন:

Month (1-12)	Day Range	Bangla M	Formula	month	Day Range	E.M	
4	14-30	1	-13	1	1-17	4	+13
5	1-14	1	+17	1	18-31	5	-17
5	15-31	2	-14	2	1-17	5	+14
6	1-14	2	+17	2	18-31	6	-17
6	15-30	3	-14	3	1-16	6	+14
7	1-15	3	+16	3	17-31	7	-16
7	16-31	4	-15	4	1-16	7	+15
8	1-15	4	+16	4	17-31	8	-16
8	16-31	5	-15	5	1-16	8	+15
9	1-15	5	+16	5	17-31	9	-16
9	16-30	6	-15	6	1-15	9	+15
10	1-15	6	+15	6	16-30	10	-15
10	16-31	7	-15	7	1-16	10	+15
11	1-14	7	+16	7	17-30	11	-16
11	15-30	8	-14	8	1-16	11	+14
12	1-14	8	+16	8	17-30	12	-16
12	15-31	9	-14	9	1-17	12	+14
1	1-13	9	+17	9	18-30	1	-17
1	14-31	10	-13	10	1-18	1	+13
2	1-12	10	+18	10	19-30	2	-18
2	13-28(29)	11	-12	11	1-16(17)	2	+12
3	1-14	11	+16	11	17(18)-30(31)	3	-16
3	15-31	12	-14	12	1-17	3	+14
4	1-13	12	+17	12	18-30	4	-17

লক্ষ করুন: ইংরেজি মাস কোন্‌টি সেটা জানতে পারলে আমরা কোন্ দু'টি তারিখের ভেতর কতো যোগ বা বিয়োগ করলে বাংল তারিখটি পাবো তা দেখানো হয়ছে। একই সময় বাংলা মাসটি কি হবে তাও উল্লেখিত আছে। অনুরূপ বাংলার ক্ষেত্রেও আমরা জানতে পারি কোন তারিখে কতোদিন যোগ বা বিয়োগ করতে হবে। একটি উদাহরণ থেকে

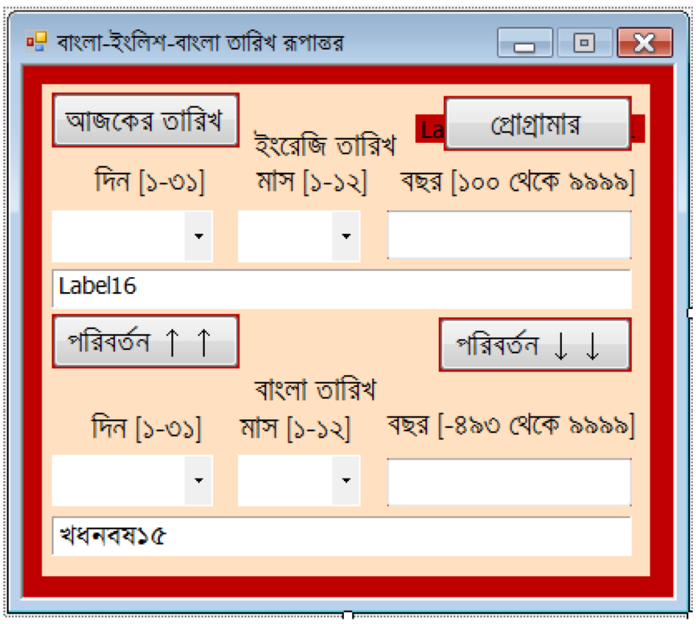
১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

বিষয়টি স্পষ্ট হবে। মনে করুন এই তারিখ: ২৫/৫/১৮৯৯ [জাতীয় কবি কাজী নজরুল ইসলামের জন্ম তারিখ] বাংলা তারিখে রূপান্তর করবো। আমরা প্রথমেই দেখবো কোন্ সংখ্যাটি বিয়োগ করে বাংলা সন বের করতে হবে। যেহেতু ইংরেজি মাসটি হলো মে [৫] যা ১৫ এপ্রিল থেকে ৩১ ডিসেম্বরের ভেতর পড়ে তাই আমাদেরকে বিয়োগ করতে হবে ৫৯৩ সংখ্যাটি। সুতরাং যে বাংলা সনে কবির জন্ম হয়েছিল তা হলো: ১৮৯৯ - ৫৯৩ = ১৩০৬ বাংলা।
এবার টেবিলের দিকে তাকিয়ে দেখতে পাচ্ছি যে, তারিখের দিনটি ইংরেজি মে (৫ম) মাসের ১৫-৩১ দিনের ভেতর হলে বাংলা তারিখের দিনটি হবে: ইংরেজি তারিখ - ১৪ = ২৫-১৪ = ১১। আর মাস হবে: দ্বিতীয়টি তথা জৈষ্ঠ্য। সুতরাং আমাদের জাতীয় কবির জন্ম হয়েছিল: ১১ জৈষ্ঠ্য, ১৩০৬ বাংলা। সুতরাং যে কোনো ইংরেজি তারিখ বাংলায় রূপান্তরের একটি সূত্র পেয়ে গেলাম। অনুরূপ বাংলা থেকেও ইংরেজিতে রূপান্তর করতে টেবিলের তথ্যাদি যথেষ্ট হবে। যেমন: আজকের [এই লেখা রচনার] তারিখ হলো ২৯ পৌষ, ১৪২৩ বাংলা। আমরা এটাও জানি, আজকের ইংরেজি তারিখ হলো ১২ জানুয়ারী, ২০১৭ ইং। আমরা টেবিল ব্যবহার করে ইংরেজি তারিখ বের করতে পারি এভাবে:
পৌষ মানে বছরের ৯তম মাস। আর তারিখটি হলো ২৯। সুতরাং টেবিলে যেয়ে দেখতে পাচ্ছি যে, ৯ম মাসের ১৮-৩০ দিনের ভেতর হলে বছর পড়ে জানুয়ারী। যেহেতু ১লা জানুয়ারী থেকে ১৩ এপ্রিল পর্যন্ত রেঞ্জের ভেতর তারিখ হলে যোগ দিতে হয় ৫৯৪ তাই ১৪২৩ বাংলা = ১৪২৩ + ৫৯৪ = ২০১৭ ইং। তারিখ বের করতেও আমরা টেবিল ব্যবহার করতে পারি। বাংলা মাস ৯ (পৌষ) ও দিন ১৮-৩০ এর ভেতর হওয়ায় বিয়োগ হবে ১৭। সুতরাং ইংরেজি দিন হবে: ২৯ - ১৭ = ১২। তাই এখন বলতে পারি: ২৯ পৌষ, ১৪২৩ বাংলা = ১২ জানুয়ারী, ২০১৭ ইং।

সুতরাং আমরা যেভাবে উপরের তারিখগুলো পরিবর্তন করেছি তা-ই করতে হবে কম্পিউটারের সফ্টওয়্যারের মাধ্যমে।

ডিজাইন: ইউজার ইন্টারফেইস ডিজাইন একটি গুরুত্বপূর্ণ কাজ। আপনি চাইবেন ব্যবহারকারী যাতে সহজে এ্যাপলিকেশনটি কাজে লাগাতে পারেন। মোটকথা প্রয়োজনীয় সুযোগ-সুবিধাসহ একটি ব্যবহারকারী-বান্ধব ইউজার ইন্টারফেইস হলো সফল এ্যাপলিকেশন সৃষ্টির চাবিকাঠি। প্রথমে নিচের ডিজাইনটি দেখুন। এরপর কি কি বস্তু টুলবাক্স থেকে আনবেন ও এদের প্রোপার্টি কি হবে তা একটি টেবিলে তুলে ধরবো।



বস্তু	নাম	টেক্সট	ফন্ট-সাইজ
ফর্ম	.....	যে কোনো	.....
৪টি বোতাম	.....	যেভাবে আছে	যেভাবে আছে
১৬টি লেবেল	.....	যেভাবে আছে	শেষের ৬টির ভিজিবল= False
৪টি কম্বোবাক্স	....	....	বাংলা- ১৪
দু'টি টেক্সট বাক্স	....	....	বাংলা - ১৪

লক্ষ করুন: উপেরর বায়ের কম্বোর নাম: cmbEnglishDay। ডানেরটির নাম: cmbEnglishMonth। নিচের বায়েরটির নাম: cmbBanglaDay। ডানেরটি: cmbBanglaMonth। উপরের টেক্সট বাক্সের নাম: txtEnglishYear। নিচেরটির নাম: txtBanglaYear। এই নামগুলো আমরা কোডে ব্যবহার করেছি। পরিবর্তন বোতামের এ্যারোগুলো আনতে সিম্বল কাজে

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

লাগান। এবার দেখে নিই কোডগুলো।

কোড: আমরা কোডের বিভিন্ন সাবরুটিন ও ফাংশন ব্যাখ্যাসহ আলাদাভাবে তুলে ধরবো। এভাবে হলে কেডগুলো বুঝা অনেকটা সহজ হবে। কারণ, তারিখ পরিবর্তন কোড বাস্তবে তেমন সহজ নয়- বরং বেশ জটিল।

ফর্মের মধ্যে ডবল ক্লিক করে কোড এডিটর খুলুন। ফর্মের ক্লাস ডিক্লারেশন লাইন: Public Class Form1 -এর নিচে লিখুন এই গ্লোবাল ভেরিয়েবল ডিক্লারেশনগুলো:

```
Dim ToBanglaAdderVal As Integer 'স্টোর করে ইংরেজি দিনের সাথে কোন্ সংখ্যা যোগ হবে
Dim ToBanglaMonth As Integer 'স্টোর করে বাংলা মাস কি হবে
Dim ToBanglaYear As Integer 'স্টোর করে বাংলা বছর মান কি হবে
Dim IsitLeapYear As Boolean 'বুলিয়ান স্টোরস লিপ ইয়ার কি না - সত্য-মিথ্যা
Dim ToEnglishAdderVal As Integer 'স্টোর করে ইংরেজি দিনের সাথে কোন্ সংখ্যা যোগ হবে
Dim ToEnglishMonth As Integer 'স্টোর করে ইংরেজি মাস কি হবে
Dim ToEnglishYear As Integer 'স্টোর করে ইংরেজি বছর কি হবে
```

উপরোক্ত ভেরিয়েবলের কোনটির কাজ কি তা কমেন্ট লাইনে লিখেছি। এবার প্রথম সাবরুটিনটি টাইপ করুন।

```
 'প্রোগ্রাম খুললেই যে তারিখ বাংলা ও ইংরেজিতে হবে
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
      PutInEnglishDate() 'সাব কল, এটা ইংরেজি আজকের তারিখ বসাবে।
      convertToBanglaDate() 'এ সাব ইংরেজি আজকের তারিখকে বাংলা করে বসাবে।
End Sub
```

আমরা এখানে দু'টি সাবরুটিন কল করেছি যেগুলো পরে ডিফাইন করা হয়েছে। এদের কাজ হলো বাংলা রূপান্তরসহ 'আজকের তারিখ' প্রদর্শন করা। সুতরাং ব্যবহারকারী প্রোগ্রাম রান করার পরই বর্তমান তারিখ ইংরেজি ও বাংলায় দেখতে পাবেন। সাবরুটিন **PutInEnglishDate()** -এর মাধ্যমে উপরের প্রথম কম্বোতে ইংরেজি দিন, পরের কম্বোতে মাস ও টেক্সট্ বাক্সে বছরের সংখ্যাটি দেখাবে। অনুরূপ নিচের প্রথম কম্বোতে বাংলা দিন, পরের কম্বোতে মাস ও টেক্সট্ বাক্সে বছরের সংখ্যা প্রদর্শন হবে দ্বিতীয় সাবরুটিন **convertToBanglaDate()** -এর মাধ্যমে। এগুলো কিভাবে কাজ করে তার ওপর আলোচনা সময়মতো হবে। পরের সাবটি হলো ইংরেজি থেকে বাংলায় পরিবর্তন বোতামের ক্লিক ইভেন্ট কোড।

```
 'এটা ইংরেজি থেকে বাংলায় পরিবর্তনের বোতাম।
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      If Val(txtEnglishYear.Text) < 100 Or Val(txtEnglishYear.Text) > 9999 Then
         MsgBox("বছর রেঞ্জ হবে ১০০ থেকে ৯৯৯৯।")
        txtEnglishYear.Text = ""
        txtEnglishYear.Focus()
        Exit Sub
      End If
      Label15.Text = ""
       'লিপ ইয়ার না হলে ফেব্রুয়ারি ২৯ এর নিচে থাকতে হবে।
       If IsItLeap(Val(txtEnglishYear.Text)) = False Then
        If (cmbEnglishDay.SelectedItem) >= 29 And (cmbEnglishMonth.SelectedItem) = 2 Then
          MsgBox("Not a leap year! February month has 28 days only.", MsgBoxStyle.Exclamation)
          Exit Sub
        End If
      Else
         'যদি লিপ ইয়ার হয়ে থাকে তাহলে ফেব্রুয়ারি অবশ্যই ২৯ কিংবা কম থাকতে হবে।
          If (cmbEnglishDay.SelectedItem) >= 30 And (cmbEnglishMonth.SelectedItem) = 2 Then
             MsgBox("Its a leap year! February month has 28 days only.", MsgBoxStyle.Exclamation)
          Exit Sub
```

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
        End If
    End If
     'লিপ ইয়ার নয় বা লিপ ইয়ার হয়, এর পরও দিনের তারিখ সঠিক থাকলে
   UpdateLabel() 'গোপন লেবেলে দিন, মাস ও বছরের মান নিয়ে যেতে হবে, এই সাবের কাজ এ  টি।
  'এরপর দেখতে হবে দিনের মাস অন্যান্য মাসের বেলায়ও সঠিক আছে কি না।
    If IsDayEntryCorrect((Val(Label10.Text)), Val(Label9.Text)) = False Then
        MsgBox("উপস্! মাসে য  দিন আছে তারচেয়ে বেশি হয়ে গেছে, বোধ হয়!", MsgBoxStyle.Exclamation)
        Exit Sub
    End If
     'এবার সব সঠিক হয়েছে
     convertToBanglaDate() 'কল কনভার্ট টু বাংলা সাবরুটিন।
End Sub
```

কোডের ব্যাখ্যা: প্রথম ইফ-এন্ড ইফ ব্লকের কাজ হলো বছরের সংখ্যাটি রেঞ্জের ভেতর আছে কি না তা নিশ্চিত করা। আমরা ইংরেজি ১০০ থেকে ৯৯৯৯ সাল পর্যন্ত রেঞ্জ দিয়েছি। এর বাইরে হলে প্রোগ্রাম মেসেজ দিয়ে কোড থেকে বের হবে। এরপর আমরা লেবেল ১৫ এর টেক্সট মুছে দিয়েছি। পরের ইফ-এন্ড ইফ ব্লকে আরেকটি টেস্ট করা হয়েছে। আমরা জানতে চাই, বছর সংখ্যাটি লিপ ইয়ার কি না। এ চেকটি করেছি একটি ফাংশন দ্বারা, যার নাম IsItLeap। এর একটিমাত্র প্যারামিটার আছে যা হলো বছর সংখ্যাটি। ফাংশন বুলিয়ান [হ্যাঁ, না] রিটার্ন করে। সুতরাং দেখতে হবে রিটার্ন করার পরও ব্যবহারকারী ফেব্রুয়ারি মাসের জন্য ২৮ দিন বা তার নিচে প্রবেশ করেছেন কি না। পরের ইফ-এন্ড ইফ ব্লকের কাজ এটি। কিন্তু লিপ ইয়ার হয়ে থাকলে দিন সংখ্যা যাতে ২৯ এর বেশি না হয় তা-ও দেখতে হবে। সুতরাং একই ইফ-এন্ড ইফ ব্লকেই Else কিওয়ার্ড দ্বারা আরো একটি চেক করতে হয়েছে। উভয় চেক সফল হওয়ার পরই পরবর্তী কোড এক্সেকিউট হতে পারে। প্রথমে আমরা একটি সাবরুটিন [`UpdateLabel()`] কল করে তারিখের সংখ্যাগুলো লুকানো তিনটি লেবেলে নিয়ে গেছি। এরপর আরো একটি সংখ্যা পরীক্ষা করা দরকার। মাসের মধ্যে য'দিন আছে তারচে বেশি প্রবেশ হয়েছে কি না তা-ও জানতে হবে। এই টেস্ট সফল হওয়ার পর আমরা অবশেষে প্রস্তুত হলাম পরিবর্তন সাবরুটিন কল করতে। এই সাবরুটিনের নাম `convertToBanglaDate()`। সঠিক সময়ে এটার ব্যাখ্যা আসবে।

ইতোমধ্যে বলেছি, আগের সাবরুটিনে `UpdateLabel()` নামক একটি সাবরুটিন কল করেছি। সুতরাং এটাই আমাদের পরবর্তী সাবরুটিন। এর কাজ হলো কম্বো বাক্স থেকে ইংরেজি দিনের সংখ্যা, ইংরেজি মাসের সংখ্যা ও [টেক্সট বাক্স থেকে] ইংরেজি বছরের সংখ্যা যথাক্রমে লেবেল ১৯, ১০ ও ১১ -তে কপি করে রাখা।

এর পরের ফাংশনটির কথাও ইতোমধ্যে বলেছি। এটা হলো `IsItLeap` নামক বুলিয়ান ফাংশন। এর কাজ হচ্ছে বছরটি লিপ ইয়ার কি না পরীক্ষা করে সত্য-মিথ্যা ফলাফল রিটার্ন করা। লক্ষ করুন, কোনো সংখ্যার ভাগফলে অবশিষ্ট আছে কি না জানতে হলে সিস্টেমের মড [Mod] ফাংশন ব্যবহার করতে হয়। সুতরাং এবার টা্ইপ করুন এ দু'টো প্রসিডোর।

```
Private Sub UpdateLabel() 'লেবেল আপডেট কোড।
      Label9.Text = cmbEnglishDay.SelectedItem 'কম্বো থেকে লেবেলে
       Label10.Text = cmbEnglishMonth.SelectedItem 'কম্বো থেকে লেবেলে
       Label11.Text = txtEnglishYear.Text 'টেক্সট বাক্স থেকে লেবেলে
End Sub
 'check if it is a Leap year
 Private Function IsItLeap(TheYear As Integer) As Boolean
      If TheYear Mod 4 = 0 Then 'মড রিটার্ন করে ভাগের অবশিষ্ট। না থাকলে ০
         Return True 'এটা লিপ ইয়ার
       Else
         Return False 'এটা লিপ ইয়ার নয়
       End If
```

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
End Function
```

উপরের সাব দু'টোর ব্যাখ্যা ইতোমধ্যে হয়ে গেছে। এবার পরের এই সাবটি লিখুন।

```
'এই সাব ইংরেজি আজকের তারিখ দিন, মাস ও বছর ফিল্ডে বসাবে।
Private Sub PutInEnglishDate()
    Dim now As DateTime = DateTime.Now
    Label9.Text = now.Day 'শুধু দিনের সংখ্যা
    Dim now1 As DateTime = DateTime.Now
    Label10.Text = now1.Month 'শুধু মাসের সংখ্যা
    Dim now2 As DateTime = DateTime.Now
    Label11.Text = now2.Year 'শুধু বছরের সংখ্যা
    cmbEnglishDay.SelectedItem = Label9.Text 'লেবেল থেকে কম্বো বাক্সে
    cmbEnglishMonth.SelectedItem = Label10.Text 'লেবেল থেকে কম্বো বাক্সে
    txtEnglishYear.Text = Label11.Text 'টেক্সট্ বাক্স থেকে লেবেলে
End Sub
```

উপরের সাবরুটিনটি আমরা ফর্মের লোড সাব থেকে কল করেছি। এটার কাজ হলো ইংরেজি 'আজকের' তারিখটিকে প্রথমে লুকানো লেবেল৯, লেবেল১০ ও লেবেল১১-তে যথাক্রমে দিন, মাস ও বছর সংখ্যা সংরক্ষণ করা ও কম্বোইংলিশডে, কম্বোইংলিশমান্থ ও টিএক্সটিইংলিশইয়ার বাক্সে প্রদর্শন করা। পরের সাবরুটিনের কাজ হলো ইংরেজি তারিখকে বাংলায় রূপান্তর করা।

```
 'এই সাবই কনভার্ট করে। কলস্ ফাংশন: ১.ফাইন্ডডেরেঞ্জ, ২. ফাইন্ডবাংলা ইয়ার।
Private Sub convertToBanglaDate()
    Dim EnglishDay As Integer = Val(Label9.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
    Dim EnglishMonth As Integer = Val(Label10.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
    Dim EnglishYear As Integer = Val(Label11.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
    Dim DayAdderMonth As String = FindDayRange(EnglishMonth, EnglishDay) 'ফাংশন কল
□□□□□Dim Words As String() = DayAdderMonth.Split(New Char() {" "})
    ToBanglaAdderVal = Words(0)
    ToBanglaMonth = Words(1)
    ToBanglaYear = FindBanglaYear(EnglishYear, EnglishMonth, EnglishDay) 'ফাংশন কল
    Label12.Text = (ToBanglaAdderVal) + EnglishDay 'এড সঠিক মান বাংলা বছর পেতে।
    Label13.Text = ToBanglaMonth 'গোপন লেবেলে মাস মান সংরক্ষণ
    cmbBanglaDay.SelectedItem = Label12.Text 'গোপন লেবেলে থেকে কম্বোতে নেওয়া
    cmbBanglaMonth.SelectedItem = Label13.Text 'গোপন লেবেলে কম্বোতে নেওয়া
    txtBanglaYear.Text = ToBanglaYear 'বাংলা বছরের ফিল্ডে মান নেওয়া
    Label16.Text = (cmbEnglishMonth.SelectedItem & "/" & cmbEnglishDay.SelectedItem & "/" &
txtEnglishYear.Text)
    Label16.Text = FormatDateTime(Label16.Text, DateFormat.LongDate) 'দিনের নামসহ তারিখ
    Label15.Text = getBanglaDayMothName(Label16.Text, Val(Label12.Text), Val(Label13.Text), Val
(txtBanglaYear.Text)) 'দিনের নামসহ তারিখ।□
End Sub
```

লক্ষ করুন, এই সাব থেকে আমরা আরো দু'টি ফাংশন কল করেছি। যথা: ১. FindDayRange এবং ২. FindBanglaYear । আশাকরি এগুলোর নাম থেকেই বুঝতে পারছেন তাদের কাজ কি। প্রথমটি রিটার্ন করে দিন একটি স্ট্রিং যার মধ্যে আছে পরিবর্তনে কোন্ সংখ্যাটি হবে এবং বাংলা মাস সংখ্যাটি। দ্বিতীয়টি রিটার্ন করে বাংলা বছর কি হবে। এই ফাংশন দু'টোর ব্যাখ্যা সঠিক স্থানে আসবে। লক্ষ করুন, আমরা Split নামক একটি সিস্টেম ফাংশন দ্বারা দু'টি সংখ্যাকে আলাদা করেছি। প্রাপ্ত দিনের মান ও মাসের মান যথাক্রমে গোপন লেবেল১২ এবং লেবেল১৩-তে নিয়ে তারপর কম্বো বাক্স দু'টোতে প্রদর্শন করেছি। অপরদিকে বছরের মানটি সরাসরি টেক্সট্‌বাক্সে নিয়ে এসেছি।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

আমাদের ইচ্ছে ব্যবহারকারী সংখ্যা হিসাবেই শুধু তারিখগুলো দেখবেন না- বরং বার ও মাসের নামসহ দেখবেন। পরের ফাংশনের কাজ হলো এটা নিশ্চিত করা। ফাংশন মোট তিনটি প্যারামিটার নেয়।

```
Public Function getBanglaDayMothName(SuplyLongDateString As String, BanglaDay As Integer, BanglaMonthVal As Integer,
BanglaYearVal As Integer) As String
    Dim AllEngDayName() As String = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"}
    Dim AllBanglaDayName() As String = {"iweevi", "†mvgevi", "g½jevi", "eyaevi", "e„n¯úwZevi", "ïµevi", "kwbevi"}
    Dim AllBanglaMonthName() As String = {"ˆekvL", "ˆRô¨", "Avlvp", "kÖveY", "fv`ª", "Avwk¦b", "KvwZ©K",
"AMÖnvqY", "†cŠl", "gvN", "dvêyb", "ˆPÎ"}
    Dim AllBangMonthsNum() As Integer = {"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12"}
    Dim RetString As String = ""
    Dim CountChar As Integer : Dim RetDay As String = "" : Dim BanglaDayname As String = ""
    Dim BanglaMonthName As String = ""
    For CountChar = 1 To Len(SuplyLongDateString)
      If Mid(SuplyLongDateString, CountChar, 1) = "," Then
        Exit For
      End If
      RetDay += Mid(SuplyLongDateString, CountChar, 1)
    Next
    For CountChar = 0 To 6
      If AllEngDayName(CountChar) = RetDay Then
        BanglaDayname = AllBanglaDayName(CountChar)
        Exit For
      End If
    Next
    For CountChar = 0 To 12
      If AllBangMonthsNum(CountChar) = BanglaMonthVal Then
        BanglaMonthName = AllBanglaMonthName(CountChar)
        Exit For
      End If
    Next
    RetString = BanglaDayname & ", " & BanglaDay & " " & BanglaMonthName & ", " & BanglaYearVal & " evsjv"
    Return RetString
  End Function
```

উপরের ফাংশন যে তিনটি প্যারামিটর নেয় এগুলো সম্পর্কে আগেই জানা দরকার। প্রথমটি হলো ইংরেজি দীর্ঘ তারিখ স্ট্রিং। এটা যে কোনো ইংরেজি তারিখ লিখে সিস্টেম দ্বারা আনা যায়। দ্বিতীয় ও তৃতীয় প্যারামিটার যথাক্রমে বাংলা দিনের সংখ্যা ও মাসের সংখ্যা। এই তিনটি ভ্যালু থেকেই বাংলা দিন ও মাসের নাম ফাংশন রিটার্ন করবে। কিভাবে? এক্ষুণি জেনে নেবেন।

প্রথমত আমরা ৪টি এ্যারে ভেরিয়েবল ডিক্লায়ার করেছি। এগুলো হয়ে: ১. AllEngDayName() যাতে আছে ইংরেজি সাত বারের নাম, ২. AllBanglaDayName() -এতে সংরক্ষিত হয়েছে বাংলা সাত বারের নাম [বিজয় কিবোর্ডে লিখেছি] ৩. AllBanglaMonthName() -এতে সংরক্ষিত আছে বাংলা বারো মাসের নাম [বিজয়ে] এবং ৪. AllBangMonthsNum() -এ ভেরিয়েবলে আছে বারো মাসের নাম্বার (১-১২)। মনে আছে এ্যারোর যাবতীয় ইলেমেন্ট সংরক্ষিত হয় ০ ইনডেক্স দ্বারা। অন্য কথায়, আমাদের প্রথম ভেরিয়েবলে আছে ০ থেকে ৬ পর্যন্ত, ইনডেক্স, দ্বিতীয়টিতেও তা-ই। তৃতীয় ও চতুর্থটিতে আছে ০ থেকে ১১ পর্যন্ত ইনডেক্স। আর আমরা এই ইনডেক্স দ্বারাই সঠিক ফলাফল বের করবো।

উপরে ব্যাখ্যাত ৪টি এ্যারে ভেরিয়েবল ছাড়াও আমরা আরো ৫টি সাধারণ ভেরিয়েবল ডিক্লায়ার করেছি ফাংশনের ভেতর কাজে লাগাতে। এদের মধ্যে ৪টির টাইপ হলো স্ট্রিং ও একটি ইন্টেজার। এগুলোর কাজ নিম্নরূপ:

১. RetString -এর কাজ বাংলা দিনের নাম, দিন সংখ্যা, মাসের নাম ও বছরের সংখ্যা একত্রে সঠিক পরম্পরায় সংরক্ষণ

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

করা। আর এটাই ফাংশন রিটার্ন করে। বাকি তিনটি স্ট্রিং ভেরিয়েবল এই প্রথমটির সহায়ক মাত্র। এগুলো দিনের নাম, মাসের নাম ও দিনের সংখ্যা সংরক্ষণ করে। ইন্টেজার ভেরিয়েবলের কাজ হলো ফর-নেক্সট লুপটি নিয়ন্ত্রণ করা।
আমরা প্রথমে একটি ফর-নেক্সট্ লুপ দ্বারা ইংরেজি মাসের নামটি বের করে সংরক্ষণ করেছি। এরপর এটির ফলাফলের ওপর ভিত্তি করে পরের ফর-নেস্ট লুপের মাধ্যমে সনাক্ত করেছি বাংলা দিনের নামটি। শেষের ফর-নেস্ট লুপ দ্বারা বের করেছি বাংলা মাসের নামটি। ব্যস! এবার সবগুলো একত্র করে রিটার্ন করতে হবে। এ কাজের জন্য ব্যবহার করেছি `RetString` ভেরিয়েবলকে। লক্ষ করুন কিভাবে ফরমেটিং করা হয়েছে। আমরা চাই প্রথমে হবে বাংলা দিনের নাম এরপর কমা ও স্পেইস। তারপর বাংলা দিনের সংখ্যা এরপর আরেকটি স্পেইস পরে আসবে বাংলা মাসের নাম- তারপর কমা ও স্পেইস শেষে বছরের সংখ্যা ও স্পেইস এবং সবশেষে 'বাংলা' শব্দটি। এই ফরমেটিং কিভাবে হয়েছে ভালো করে দেখে নেবেন। এরপর এসছে দু'টি বড়ো ফাংশনের প্রথমটি।

```
'function returns bangla day and Month to be added or subtracted
 Private Function FindDayRange(MonthInt As Integer, DayInt As Integer) As String
   If MonthInt = 1 And DayInt >= 1 And DayInt <= 13 Then
     Return "17" & " 9"
   End If
   If MonthInt = 1 And DayInt >= 14 And DayInt <= 31 Then
     Return "-13" & " 10"
   End If
   If MonthInt = 2 And DayInt >= 1 And DayInt <= 12 Then
     Return "18" & " 10"
   End If
   If MonthInt = 2 And DayInt >= 13 And DayInt <= 29 Then
     Return "-12" & " 11"
   End If
   If MonthInt = 3 And DayInt >= 1 And DayInt <= 14 Then
     Return "16" & " 11"
   End If
   If MonthInt = 3 And DayInt >= 15 And DayInt <= 31 Then
     Return "-14" & " 12"
   End If
   If MonthInt = 4 And DayInt >= 1 And DayInt <= 13 Then
     Return "17" & " 12"
   End If
   If MonthInt = 4 And DayInt >= 14 And DayInt <= 30 Then
     Return "-13" & " 1"
   End If
   If MonthInt = 5 And DayInt >= 1 And DayInt <= 14 Then
     Return "17" & " 1"
   End If
   If MonthInt = 5 And DayInt >= 15 And DayInt <= 31 Then
     Return "-14" & " 2"
   End If
   If MonthInt = 6 And DayInt >= 1 And DayInt <= 14 Then
     Return "17" & " 2"
   End If
   If MonthInt = 6 And DayInt >= 15 And DayInt <= 30 Then
     Return "-14" & " 3"
   End If
   If MonthInt = 7 And DayInt >= 1 And DayInt <= 15 Then
     Return "16" & " 3"
   End If
   If MonthInt = 7 And DayInt >= 16 And DayInt <= 31 Then
```

ব্যাখ্যা: এই ফাংশনটি বেশ বড়ো সত্যি, কিন্তু এটি একটি মাত্র স্ট্রিং ভ্যালু রিটার্ন করে, যার মধ্যে আছে আমাদের কাঙ্ক্ষিত দু'টি সংখ্যা। ফাংশন দু'টি প্যারামিটার গ্রহণ করে। এগুলো হলো: ইংরেজি মাসের সংখ্যা [১-১২] এবং দিনের সংখ্যা [১-৩১]। এ দু'টো সংখ্যা কি কি তার ওপর নির্ভর করে বাংলা দিনের সংখ্যা বের করতে কোন্ সংখ্যাটি যোগ বা বিয়োগ দিতে হবে এবং বাংলা মাসের সংখ্যাটি কি হবে। যেমন মাস জানুয়ারী [১] হলে এবং দিন ১ থেকে ১৩ হলে বাংলা দিন সংখ্যা বানাতে ১৭ যোগ দিতে হবে। অনুরূপ মাস সংখ্যা যদি ৪ হয় আর দিনের সংখ্যা ১৪ থেকে ৩০ থাকে, তাহলে বাংলা দিন বানাতে বিয়োগ দিতে হবে ১৩ ইত্যাদি। ফাংশন একে একে সবগুলো কম্বিনেশন চেক করে সঠিক যোগ বা বিয়োগের সংখ্যাটি ও মাসের মানটি কলিং সাবে রিটার্ন করে। পূর্বে আমরা যে সম্পর্কের টেবিলটি দেখেছিলাম সে'টি কাজে লাগিয়েই এই ফাংশনটি তৈরি করেছি। লক্ষ করুন, বিয়োগ বুঝাতে সংখ্যার পূর্বে একটি বিয়োগ চিহ্ন দেওয়াই যথেষ্ট।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
        Return "-15" & " 4"
    End If
    If MonthInt = 8 And DayInt >= 1 And DayInt <= 15 Then
        Return "16" & " 4"
    End If
    If MonthInt = 8 And DayInt >= 16 And DayInt <= 31 Then
        Return "-15" & " 5"
    End If
    If MonthInt = 9 And DayInt >= 1 And DayInt <= 15 Then
        Return "16" & " 5"
    End If
    If MonthInt = 9 And DayInt >= 16 And DayInt <= 30 Then
        Return "-15" & " 6"
    End If
    If MonthInt = 10 And DayInt >= 1 And DayInt <= 15 Then
        Return "15" & " 6"
    End If
    If MonthInt = 10 And DayInt >= 16 And DayInt <= 31 Then
        Return "-15" & " 7"
    End If
    If MonthInt = 11 And DayInt >= 1 And DayInt <= 14 Then
        Return "16" & " 7"
    End If
    If MonthInt = 11 And DayInt >= 15 And DayInt <= 30 Then
        Return "-14" & " 8"
    End If
    If MonthInt = 12 And DayInt >= 1 And DayInt <= 14 Then
        Return "16" & " 8"
    End If
    If MonthInt = 12 And DayInt >= 15 And DayInt <= 31 Then
        Return "-14" & " 9"
    End If
    Return 0
End Function
```

ইতোমধ্যে যে ফাংশনটির ওপর বিস্তারিত বলা হয়েছে তা থেকে আমরা বাংলা দিন ও মাস পাবো মাত্র। তবে আমাদেরকে তো বাংলা বছর কি হবে তা-ও জানতে হবে। এটা জানার জন্য পরের এই ফাংশনটি।

```
Private Function FindBanglaYear(EngYear As Integer,
EngMonth As Integer, DateDay As Integer) As Integer
    If EngMonth >= 1 And EngMonth <= 3 Then
        ToBanglaYear = EngYear - 594 'because
banglaYear 1 less, no Boishak yet
        Return ToBanglaYear
    End If
    If EngMonth = 4 And DateDay >= 1 And
DateDay <= 13 Then
        ToBanglaYear = EngYear - 594 'because
banglaYear 1 less, no Boishak yet
        Return ToBanglaYear
    End If
    ToBanglaYear = EngYear - 593 'Here it change
    Return ToBanglaYear
End Function
```

লক্ষ করুন, উপরের ফাংশনে তিনটি প্যারামিটার আছে। এগুলো হলো যথাক্রমে ইংরেজি বছর, মাস ও দিন। প্রতিটি ভ্যালু হবে ইন্টেজার। ফাংশন রিটার্ন করবে বাংলা বছর [ইন্টেজার হিসাবে]। আমরা জানি ১লা বৈশাখ না আসা পর্যন্ত বাংলা বছর সংখ্যা বাড়বে না। এ কারণেই প্রথমে ইংরেজি মাস ১ থেকে ৩ এর ভেতর হলে বাংলা মাস বিয়োগ করেছি ৫৯৪ সংখ্যা দ্বারা। বাকি মাসগুলোর ক্ষেত্রে বিয়োগ হবে ৫৯৩।

আমাদের ফর্মে উপরের বায়ের বোতামটি হলো বর্তমান তারিখ বাংলা ও ইংরেজি তারিখ প্রর্দনের বোতাম। সুতরাং এটার ক্লিক ইভেন্ট হবে নিম্নরূপ।

```
'English to Bangla Date conversion button sub
Private Sub Button2_Click(sender As Object, e As
EventArgs) Handles Button2.Click
    Label15.Text = ""
    PutInEnglishDate() 'পরিবর্তন [ইংরেজি থেকে বাংলা]
বোতাম। কল সাব।
    convertToBanglaDate() 'কল সাব পরিবর্তন করতে।
End Sub
```

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

এই সাবে তিনটি মাত্র লাইন আছে। প্রথমে মুছে দিয়েছি লেবেল১৫। এ লেবেলে নতুন লেখা আসবে। এর পরের দু'টি লাইন হলো দু'টি সাবরুটিন। এ উভয়টির ব্যাখ্যা ইতোমধ্যে হয়েছে। পরের ফাংশনের কাজ হলো ইংরেজি দিন সংখ্যাটি মাসের সাথে সামঞ্জস্যশীল আছে কি না তা পরীক্ষা করা। মনে করুন ব্যবহারকারী মাস ৯ [সেপ্টেম্বট] প্রবেশ করলেন। এখন দিনের ফিল্ডে যদি ৩১ প্রবেশ করেন তাহলে তো সমস্যা দাঁড়াবে। কারণ, সেপ্টেম্বর মাস ৩০ দিনের বেশি হয় না। এই সমস্যা নিরসনে নিচের ফাংশনটি কাজ করে। মাসের মান অনুযায়ী দিনের মান না থাকলে ফাংশন মিথ্যা ফেরৎ পাঠাবে। অন্যথায় সত্য।

```
 'Day entry correct: Returns True, Else False
Private Function IsDayEntryCorrect(Month As Integer, Day As Integer) As Boolean
    If Month = 9 And Day > 30 Then
      Return False
    End If
    If Month = 4 And Day > 30 Then
      Return False
    End If
    If Month = 6 And Day > 30 Then
      Return False
    End If
    If Month = 11 And Day > 30 Then
      Return False
    End If
    Return True
End Function
```

এখন আমরা বাংলা থেকে ইংরেজি তারিখ পরিবর্তন বোতামের ক্লিক ইভেন্ট কোড দেখবো।

```
'Bangla to English Date conversion button sub
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    If Val(txtBanglaYear.Text) < -493 Or Val(txtBanglaYear.Text) > 9999 Then
       MsgBox("বছর রেঞ্জ হবে -৪৯৩ থেকে ৯৯৯৯।", MsgBoxStyle.Exclamation) 'রেঞ্জের বাইরে।
      txtBanglaYear.Text = ""
      txtBanglaYear.Focus()
      Exit Sub
    End If
    Label15.Text = ""
    UpdateBLabel() 'take day, month and year values to hidden labels
    If IsBanglaDateCorrect(Val(Label12.Text), Val(Label13.Text), Val(Label14.Text)) = False Then
       MsgBox("উপস্ তারিখ সঠিক হয় নি! যে মাসে যতোদিন আছে তারচেয়ে বেশি হয়েছে, সম্ভবত।",
MsgBoxStyle.Exclamation)
    End If
    convertToEnglishDate()
End Sub
```

প্রথমে বছরের রেঞ্জ সঠিক আছে কি না পরীক্ষা করে [সঠিক থাকলে] আমরা UpdateBLabel() সাব কল করে দিন, মাস ও বছরের মানগুলো গোপন লেবেলে নিয়ে সংরক্ষণ করেছি। এরপর আবারো তরিখ সঠিক কি না পরীক্ষা করে [সঠিক হলে] ডাক দিয়েছি পরিবর্তন করার মূল সাবরুটিন convertToEnglishDate()। পরের সাবটি হলো UpdateBLabel() -এর সাবরুটিন। এটার কাজ কি তা ইতোমধ্যে বলেছি।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
'update Bagla label
Private Sub UpdateBLabel() 'লেবেল আপডেট কোড।
        Label12.Text = cmbBanglaDay.SelectedItem 'কম্বো থেকে লেবেলে
        Label13.Text = cmbBanglaMonth.SelectedItem 'কম্বো থেকে লেবেলে
        Label14.Text = txtBanglaYear.Text 'টেক্সট বাক্স থেকে লেবেলে
End Sub
```

আমরা ইতোমধ্যে বলেছি, বাংলা তারিখ ঠিক আছে কি না পরীক্ষার জন্য একটি ফাংশন আছে। নিচের কোডগুলোই হচ্ছে এ ফাংশন। এটি লিপ ইয়ার ও সঠিক তারিখ, মাসের সঙ্গে তারিখের সামঞ্জস্য ইত্যাদি পারীক্ষা করে সত্য বা মিথ্যা ফলাফল হিসাবে রিটার্ন করে।

```
'check to see if the dateEntry correct
Private Function IsBanglaDateCorrect(BDay As Integer, BMonth As Integer, BYear As Integer) As Boolean
        Dim CheckLeap As Boolean = IsItLeap(BYear)
        If CheckLeap = True Then
            If BMonth = 11 And BDay <= 31 Then
                Return True
            End If
            If BMonth >= 1 And BMonth <= 5 Then
                If BDay <= 31 Then
                    Return True
                End If
            End If
            If BMonth >= 6 And BMonth <= 10 Then
                If BDay <= 30 Then
                    Return True
                End If
            End If
            If BMonth = 12 And BDay <= 30 Then
                Return True
            End If
        End If
        'leap year false
        If BMonth >= 1 And BMonth <= 5 Then
            If BDay <= 31 Then
                Return True
            End If
        End If
        If BMonth >= 6 And BMonth <= 11 Then
            If BDay <= 30 Then
                Return True
            End If
        End If
        If BMonth = 12 And BDay <= 30 Then
            Return True
        End If
        Return False
End Function
```

পরের সাবরুটিনের কাজ হলো বাংলা তারিখকে ইংরেজি তারিখে রূপান্তর করা।

১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

```
'bangla to english date conversion
Private Sub convertToEnglishlDate()
    Dim BanglaDay As Integer = Val(Label12.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
    Dim BanglaMonth As Integer = Val(Label13.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
    Dim BanglaYear As Integer = Val(Label14.Text) 'গোপন লেবেল থেকে সংখ্যা নিয়ে আসা
    Dim DayAdderMonth As String = FindEnglishRange(BanglaMonth, BanglaDay) 'ফাংশন কল
    Dim Words As String() = DayAdderMonth.Split(New Char() {" "})
    ToEnglishAdderVal = Words(0)
    ToEnglishMonth = Words(1)
    ToEnglishYear = FindEnglishYear(BanglaYear, BanglaMonth, BanglaDay) 'ফাংশন কল
    Label9.Text = BanglaDay + (ToEnglishAdderVal)  'এড সঠিক মান বাংলা দিন পেতে।
    Label10.Text = ToEnglishMonth 'গোপন লেবেলে মাস মান সংরক্ষণ
    cmbEnglishDay.SelectedItem = Label9.Text 'গোপন লেবেল থেকে কম্বোতে নেওয়া
    cmbEnglishMonth.SelectedItem = Label10.Text 'গোপন লেবেল থেকে কম্বোতে নেওয়া
    txtEnglishYear.Text = ToEnglishYear 'বাংলা বছরের ফিল্ডে মান নেওয়া
    Label16.Text = cmbEnglishMonth.SelectedItem & "/" & Label9.Text & "/" & txtEnglishYear.Text
    Label16.Text = FormatDateTime(Label16.Text, DateFormat.LongDate)
    Label15.Text = getBanglaDayMothName(Label16.Text, Val(Label12.Text), Val(Label13.Text), Val
    (Label14.Text)) 'দিনের নামসহ তারিখ
End Sub
```

এই সাবরুটিন দু'টি গুরুত্বপূর্ণ ফাংশন কল করে। এগুলো সম্পর্কে ব্যাখ্যা একটু পরই আসছে। এখানে যা বলার তাহলো, প্রথম ফাংশন রিটার্ন করে একটি স্ট্রিংয়ের মধ্যে দু'টি মান। প্রথমটি হলো ইংলিশ দিনের সাথে কতো যোগ বা বিয়োগ করতে হবে। আর দ্বিতীয়টি হলো বাংলা মাসটি কি হবে। আমরা স্ট্রিংকে সিস্টেমের Split নামক ফাংশন দ্বারা আলাদা দু'টি শব্দে বিভক্ত করে মানগুলো বের করে যথাক্রমে ToEnglishAdderVal এবং ToEnglishMonth ভেরিয়েবলে সংরক্ষণ করেছি। দ্বিতীয় ফাংশন থেকে প্রাপ্ত ইংলিশ বছরের মান সরাসরি টেক্সবাক্সে নিয়ে গেছি প্রদর্শনের জন্য। লক্ষ করুন টেক্সট বাক্স ১৫ ও ১৬-তে দিনের ও মাসের নামসহ যথাক্রমে ইংরেজি ও বাংলা তারিখটি বসিয়েছি। ঠিক যেমনটি করেছিলাম ইংরেজি থেকে বাংলায় তারিখ পরিবর্তনের সময়। এবার আরো একটি ফাংশনের কোড দেখবো।

```
'returns bangla year
Private Function FindEnglishYear(BangYear As Integer, BangMonth As Integer, DateDay As Integer) As Integer
    If BangMonth >= 1 And BangMonth <= 8 Then
        ToEnglishYear = BangYear + 593 'because banglaYear 1 less, no Boishak yet
        Return ToEnglishYear
    End If
    If BangMonth = 9 And DateDay >= 1 And DateDay <= 17 Then
        ToEnglishYear = BangYear + 593 'because banglaYear 1 less, no Boishak yet
        Return ToEnglishYear
    End If
    ToEnglishYear = BangYear + 594 'Here it change
    Return ToEnglishYear
End Function
```

আমাদেরকে ইংরেজি বছরের মান বের করতে হবে। উপরের ফাংশনের কাজ এটি। লক্ষ করুন, বাংলা মাস যদি ১ থেকে ৪ এর ভেতর হয় তাহলে যোগ করতে হবে ৫৯৩। একইভাবে বাংলা মাস ৯ [ফাল্গুন] হলে ও দিনের মান ১ থেকে ১৭ পর্যন্ত হলে যোগ হবে ৫৯৩। অন্যথায় যোগ হবে ৫৯৪।

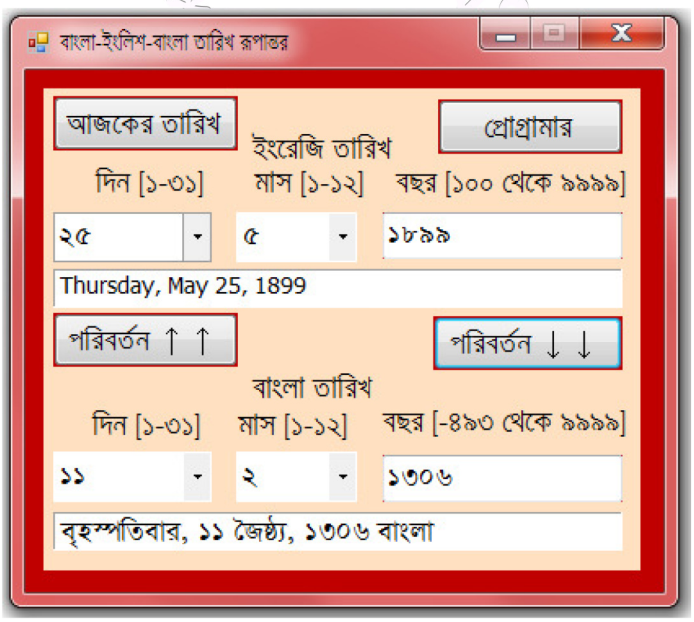
১৬তম পরিচ্ছেদ: *Few More Sample Applications*

আরো কয়েকটি নমুনা এ্যাপলিকেশন: *Few More Sample Applications*

এবার যে ফাংশনটি দেখবো তা-ই প্রোগ্রামের সর্বশেষ কোড ব্লক। অবশ্য ডানের উপরের বোতামের কোডও আরেকটি আছে। এটা প্রোগ্রামার সম্পর্কে জ্ঞাতব্য ব্যাপার মাত্র।

```
'returns what number to be added [or subtracted] to get correct english day value
Private Function FindEnglishRange(MonthInt As Integer, DayInt As Integer) As String
    If MonthInt = 1 And DayInt >= 1 And DayInt <= 17 Then
      Return "13" & " 4"
    End If
    If MonthInt = 1 And DayInt >= 18 And DayInt <= 31 Then
      Return "-17" & " 5"
    End If
    If MonthInt = 2 And DayInt >= 1 And DayInt <= 17 Then
      Return "14" & " 5"
    End If
    If MonthInt = 2 And DayInt >= 18 And DayInt <= 31 Then
      Return "-17" & " 6"
    End If
    If MonthInt = 3 And DayInt >= 1 And DayInt <= 16 Then
      Return "14" & " 6"
    End If
    If MonthInt = 3 And DayInt >= 17 And DayInt <= 31 Then
      Return "-16" & " 7"
    End If
    If MonthInt = 4 And DayInt >= 1 And DayInt <= 16 Then
      Return "15" & " 7"
    End If
    If MonthInt = 4 And DayInt >= 17 And DayInt <= 31 Then
      Return "-16" & " 8"
    End If
    If MonthInt = 5 And DayInt >= 16 And DayInt <= 15 Then
      Return "15" & " 8"
    End If
    If MonthInt = 5 And DayInt >= 17 And DayInt <= 31 Then
      Return "-16" & " 9"
    End If
    If MonthInt = 6 And DayInt >= 1 And DayInt <= 15 Then
      Return "15" & " 9"
    End If
    If MonthInt = 6 And DayInt >= 16 And DayInt <= 30 Then
      Return "-15" & " 10"
    End If
    If MonthInt = 7 And DayInt >= 1 And DayInt <= 16 Then
      Return "15" & " 10"
    End If
    If MonthInt = 7 And DayInt >= 17 And DayInt <= 30 Then
      Return "-16" & " 11"
    End If
    If MonthInt = 8 And DayInt >= 1 And DayInt <= 16 Then
      Return "14" & " 11"
    End If
    If MonthInt = 8 And DayInt >= 17 And DayInt <= 30 Then
      Return "-16" & " 12"
    End If
    If MonthInt = 9 And DayInt >= 1 And DayInt <= 17 Then
      Return "14" & " 12"
    End If
    If MonthInt = 9 And DayInt >= 18 And DayInt <= 30 Then
      Return "-17" & " 1"
    End If
    If MonthInt = 10 And DayInt >= 1 And DayInt <= 18 Then
      Return "13" & " 1"
    End If
    If MonthInt = 10 And DayInt >= 19 And DayInt <= 30 Then
      Return "-18" & " 2"
    End If
    If MonthInt = 11 And DayInt >= 1 And DayInt <= 17 Then
      Return "12" & " 2"
    End If
    If MonthInt = 11 And DayInt >= 17 And DayInt <= 31 Then
      Return "-16" & " 3"
    End If
    If MonthInt = 12 And DayInt >= 1 And DayInt <= 17 Then
      Return "14" & " 3"
    End If
    If MonthInt = 12 And DayInt >= 18 And DayInt <= 30 Then
      Return "-17" & " 4"
    End If
    Return 0
End Function
```

এখানেই ফাংশনের শেষ। এ'টি কাজ করে আগে বর্ণিত FindDayRange নামক ফাংশনের মতো। ফাংশনের দু'টি প্যারামিটার আছে। তাহলো, ইংরেজি মাসের মান ও দিনের মান। কোন্ মাস ও কোন দিন তা-ই নির্দিষ্ট করে বাংলা কোন্ মাস ও দিন হবে। আগেরটির মতো এই ফাংশনও একটি স্ট্রিং রিটার্ন করে যেখানে দু'টি সংখ্যা থাকবে। আমরা কলিং সাবরুটিনে Split নামক সিস্টেম ফাংশন দ্বারা আলাদা করেছি। এরপর উভয় সংখ্যা কাজে লাগিয়েছি। যাক, কোডের ব্যাখ্য এখানেই শেষ। এবার রান করে দেখুন [নিচের চিত্র]।



১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

আপনি নিশ্চয়ই এ গ্রন্থ অধ্যয়নকালে Visual Studio Express For Desktop [2008, 2010, 2012, 2013 or 2015] এর যে কোনো একটি ব্যবহার করছেন । ভার্শন যেটাই ব্যবহার করুন না কেনো, গ্রন্থে উল্লেখিত সকল কোডই যে কোনোটির জন্য প্রজোয্য। তবে আমি কোডগগুলো লিখেছি ভার্শন ২০১২ দ্বারা। প্রোগ্রামারদের সুবিধার জন্য মাইক্রসফট উক্ত প্রতিটি ভার্শনের মধ্যেই [সামান্য রদবদলসহ] একটি উন্নতমানের টুল্স বাক্স সম্পৃক্ত করেছেন। আপনারা দেখেছেন এ টুলস্বাক্স থেকে কিভাবে আমার বোতাম, টেক্সটবাক্স, কম্বোবাক্স ইত্যাটি বস্ত ডবলক্লিক বা টেনে টেনে ফর্মে নিয়ে এসেছি। এরপর এদের প্রোপার্টি নিজের চাহিদামাফিক পরিবর্তন করেছি। এই টুল্স বাক্সের মধ্যে যা কিছু আছে তার সবগুলো আমার ব্যবহার কখনো করি নি। তবে এগুলোর মৌলিক কার্যকারিতা নিয়ে এ পর্যন্ত বিস্তারিত আলোচনা হয় নি। এ পরিচ্ছেদ রচনার উদ্দেশ্য হলো টুল্স বাক্সের বস্ত বা অবজেক্টের উপর তথ্যাদি তুলে ধরা। প্রয়োজনে ছোট ছোট কোড্ উদাহরণও থাকবে। আমি উপদেশ দেবো, আপনি প্রথমে একটি উইন্ডোজ ফর্ম প্রজেক্ট তৈরি করুন। ফর্মের আয়তন বড়ো করুন। যেসব কোড নিয়ে আলোচনা হবে ওগুলো বাস্তবে লিখে প্রোগ্রাম চালিয়ে দেখুন- এতে কোড ও বস্তুটি সম্পর্কে অতিরিক্ত জানা হয়ে যাবে।

যাক, প্রথমে আমরা টুল্স বাক্সের পুরো অবজেক্ট লিস্ট তুলে ধরছি। এরপর একে একে ব্যাখ্যা হবে।

1. Pointer
2. BackgroundWorker
3. BindingNavigator
4. BindingSource
5. Button
6. CheckBox
7. CheckedListBox
8. ColorDialog
9. ComboBox
10. ContextMenuStrip
11. DataGridView
12. DataSet
13. DateTimePicker
14. DirectoryEntry
15. DirectorySearcher
16. DomainUpDown
17. ErrorProvider
18. EventLog
19. FileSystemWatcher
20. FontDialog
21. GroupBox
22. HelpProvider
23. HScrollBar
24. ImageList
25. Label
26. LinkLabel
27. ListBox
28. ListView
29. MaskedTextBox
30. MenuStrip
31. MessageQueue
32. MonthCalender
33. NotifyIcon
34. NumericUpDown
35. OpenFileDialog
36. PageSetupDialog
37. Panel
38. PerformanceCounter
39. PictureBox
40. PrintDialog
41. PrintDocument
42. PrintPreviewControl
43. PrintPreviewDialog
44. Process
45. ProgressBar
46. PropertyGrid
47. RadioDutton
48. RicjTextBox
49. SaveFileDialof
50. SerialPort
51. ServiceController
52. SplitContainer
53. Splitter
54. StatusStrip
55. TabControl
56. TableLayoutPanel
57. TextBox
58. Timer
59. ToolStrip
60. ToolStripContainer
61. ToolTip
62. TrackBar
63. TreeView
64. VScrollBar
65. WebBrowser
66. Chart [Inside Data Sub Folder]

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

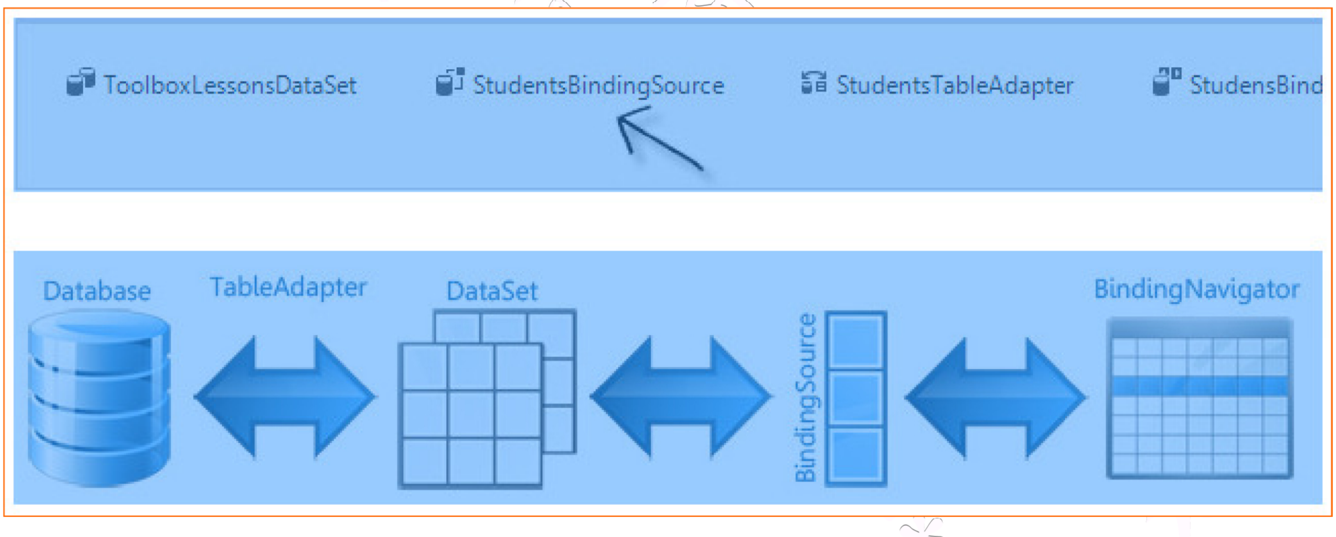
1. Pointer: টুল্‌স বাক্সের প্রথম সদস্য হলেও এটা বাস্তবে কোনো কন্ট্রোল নয়! তবে ফর্মে কন্ট্রোল নেওয়ার জন্য এটি কাজ করে। এছাড়া সিলেক্ট করা কোনো কন্ট্রোলকে ডিসিলেক্ট করতেও এটি ব্যবহৃত হয়।

2. BackgroundWorker: এটা বেকগ্রাউন্ড বা নেপথ্যে কাজ করে- কিন্তু ফর্মের ওপর এর কোনো এফেক্ট নেই। অথচ ফর্মের কোড থেকেই একে নির্দেশনা দেওয়া যায়। দীর্ঘ কোনো অপারেশনে বেকগ্রাউন্ডওয়ার্কার খুব ব্যবহারোপযোগী।

3. BindingNavigator: এটা ডাটা রেকর্ডের মধ্যে ন্যাভিগেট করার জন্য ব্যবহৃত হয়। আমরা যখন ভিজুয়্যাল স্টুডিও এর উইজার্ড দ্বারা ডাটাবেইজ প্রোগ্রাম সৃষ্টি করেছিলাম, তখন এমনিতেই উপরে কয়েকটি ন্যাভিগেশন বোতাম এসেছিল। এগুলো এই বাইন্ডিংন্যাভিগেটরের কাজ। এটি ফর্মের নিচে ধূসর এলাকায় পড়ে থাকে। চিত্রটি দেখুন।

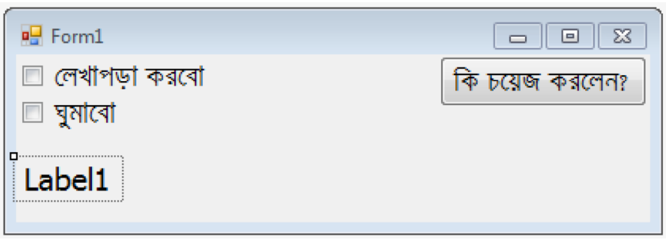


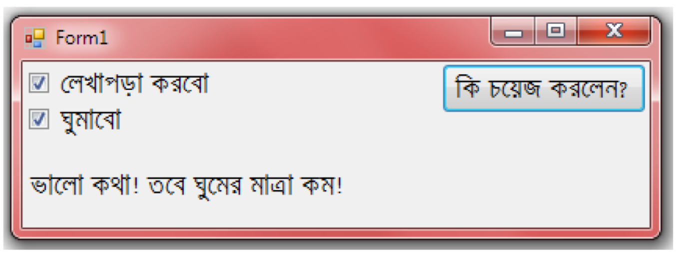
4. BindingSource: এটার কাজ হলো ফর্মের মধ্যে সম্পর্কিত ডাটা নিয়ন্ত্রণ করা। ভিজুয়্যাল বেসিক.নেট ডাটাবেইজের ডাটা প্রদর্শন ও নিয়ন্ত্রণে সাধারণত চারটি অবজেক্ট বা কন্ট্রোল ব্যবহার করে: 1. DataSet, 2. TableAdapter, 3. BindingSource & 4. BindingNavigator। দেখুন নিচের ডায়াগ্রামটি।



5. Button: এটা সম্পর্কে বেশি বলার প্রয়োজন নেই- শুধু এটুকু ছাড়া: প্রায় সকল এ্যাপলিকেশনেই এই অবজেক্টটি ব্যবহৃত হয়। এ গ্রন্থের প্রতিটি উদাহরণে এক বা ততোধিক বোতাম ব্যবহৃত হয়েছে।

6. CheckBox: এ বস্তুটি আমরা খুব বেশি ব্যবহার করি নি। এটা মূলত একটি বাক্স যেখানে ব্যবহারকারী ক্লিক করলে একটি চেকমার্ক আসে। পুনরায় ক্লিক করলে তা চলে যায়। বাক্সের ডানে [আপনার নিজস্ব] যা লেখা থাকবে তা অত্যন্ত গুরুত্বপূর্ণ। যেমন আপনি লিখতে পারেন: Volume এবং আরেকটিতে লিখতে পারেন: Area। এখন ব্যবহারকারী উভয়টি কিংবা যে কোনো একটি কিংবা কোনোটি নয়- এভাবে চেকমার্ক করতে পারেন। এ থেকে আপনি জানলেন কোন্‌টি চয়েজ করা হয়েছে ও প্রয়োজনীয় একশন নিলেন কোডের মাধ্যমে। একটি ছোট্ট উদাহরণ দেখুন। বায়ের চিত্রে

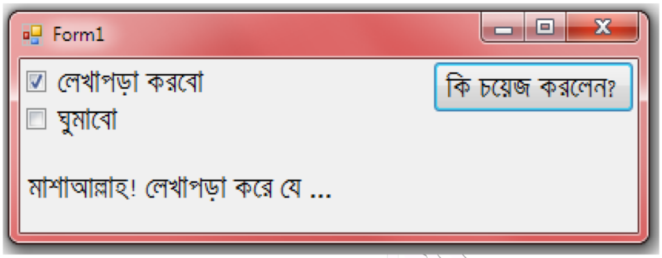


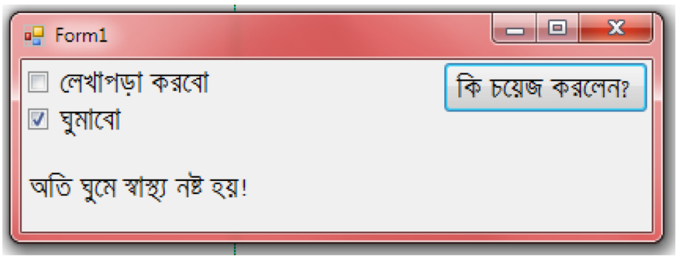


১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

ডিজাইন তুলে ধরেছি ডানেরটিতে উভয় বোতাম চয়েজ করার পর কী মেসেজ লেবেলে আসবে তা দেখা যাচ্ছে। বোতামের কোডে আরো দু'টি চয়েজের জন্য দু'টি ভিন্ন মেসেজও আছে। নিচের চিত্রদ্বয় দেখুন।



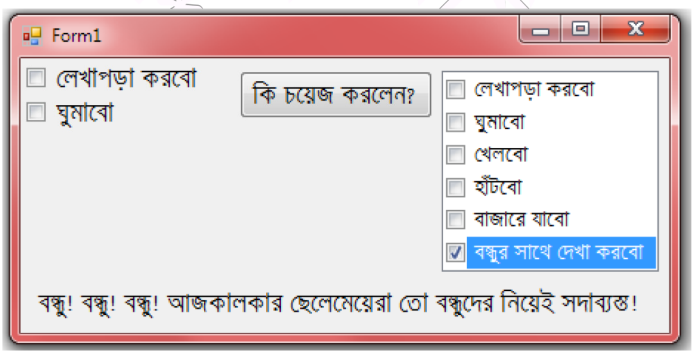


ফর্মের 'কি চয়েজ করলেন?' বোতামের ক্লিক ইভেন্ট সাবে লিখুন:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
      If CheckBox1.Checked = True Then
         Label1.Text = "মাশাআল্লাহ! লেখাপড়া করে যে ..."
      End If
      If CheckBox2.Checked = True Then
         Label1.Text = "অতি ঘুমে স্বাস্থ্য নষ্ট হয়!"
      End If
      If CheckBox1.Checked = True And CheckBox2.Checked = True Then
         Label1.Text = "ভালো কথা! তবে ঘুমের মাত্রা কম!"
      End If
      If CheckBox1.Checked = False And CheckBox2.Checked = False Then
         Label1.Text = "আরে! ঘুমাবেনও না, পড়বেনও না! মনে হয় আপনি লেজি!!"
□□□□□□□□End If
End Sub
```

7. CheckedListBox: এটা উপরে CheckBox ব্যাখ্যাকৃত কন্ট্রোলের মতোই কাজ করে। তবে অবশ্যই কয়েকটি মৌলিক পার্থক্য আছে। একটি উদাহরণ দেখি।

ফর্মের মধ্যে একটি CheckedListBox control যোগ করুন। কন্ট্রোলের উপর রাইট ক্লিক করে চয়েজ করুন: Edit Items ...। একটি জানালা আসবে এতে লিস্ট লিখুন:



□&(□□) □□ □□□□
*□□□□□
□&(□□□
□□+ □□□□□□□□
□, □□$ □□+ □□' (□□ □□□□
□

Ok বোতামে ক্লিক করুন। দেখবেন লিস্টটি এসে গেছে।

এবার নিচের কোড ব্লকটি চেকড্‌লিস্টবাক্সের মধ্যে ক্লিক করার পর যে সাবরুটিন আসবে তার ভেতর লিখুন। রান করুন, টেস্ট করুন।

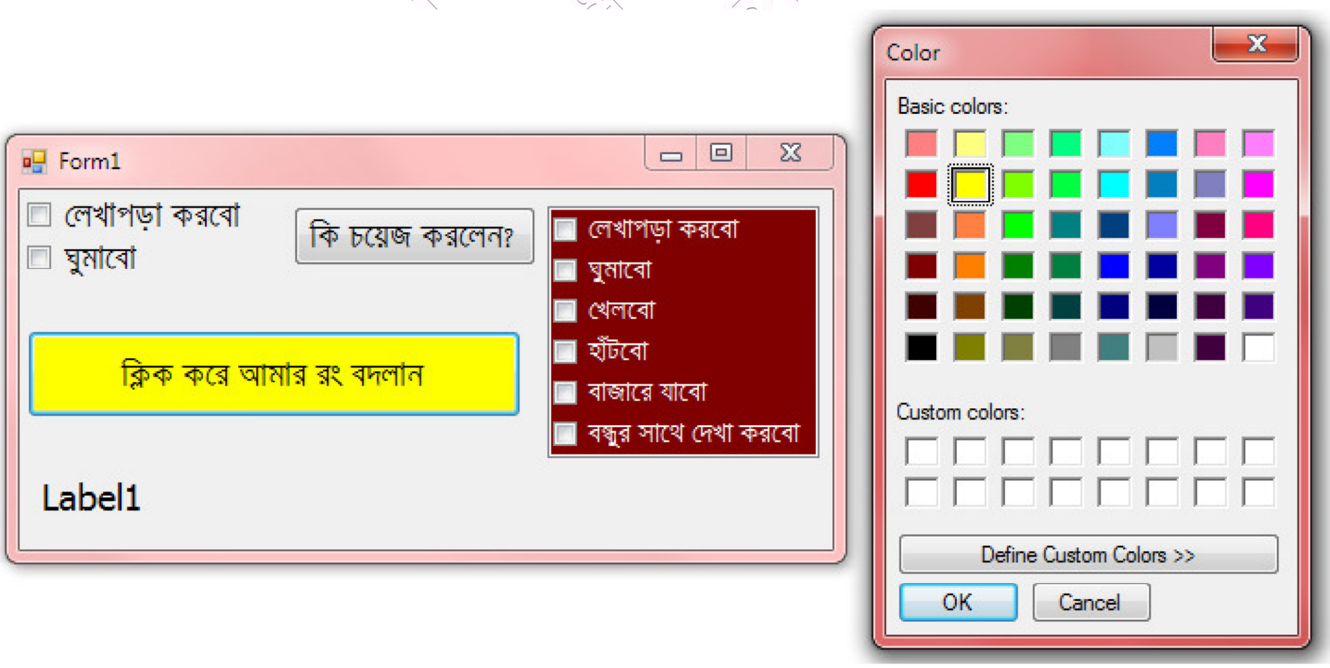
```
Private Sub CheckedListBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
CheckedListBox1.SelectedIndexChanged
      With CheckedListBox1
        If .SelectedIndex = 0 Then
           Label1.Text = "... গাড়ি ঘোড়া চড়বে! জ্ঞানী হবে।"
              End If
        If .SelectedIndex = 1 Then
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
                Label1.Text = "ঠিক আছে! সাবইকেই ঘুমোতে হয়- তবে অতিরিক্ত নয়!"
            End If
            If .SelectedIndex = 2 Then
                Label1.Text = "খেলাধূলো শরীরের জন্য ভালো বটে! কিন্তু বেশি না!"
            End If
            If .SelectedIndex = 3 Then
                Label1.Text = "অকে! হাঁটলে ব্যায়াম হয়। ব্যায় হলে স্বাস্থ্য ভালো থাকে!"
            End If
            If .SelectedIndex = 4 Then
                Label1.Text = "খুব তাড়াতাড়ি ফিরে এসো! বধুয়া অপেক্ষা করিতেছে, রাঁধিবে!"
            End If
            If .SelectedIndex = 5 Then
                Label1.Text = "বন্ধু! বন্ধু! বন্ধু! আজকালকার ছেলেমেয়েরা তো বন্ধুদের নিয়েই সদাব্যস্ত!"
            End If
        End With
End Sub
```

8. ColorDialog: এটা হচ্ছে রং করার ডাইলোগ প্রদর্শনের কন্ট্রোল। নিচের উদাহরণটি দেখুন। উপরের ফর্মে একটি বোতাম ও একটি কালার ডায়লগ কন্ট্রোল টুলবাক্স থেকে যোগ করুন। বোতামের ক্লিক ইভেন্ট সাবরুটিনে লিখুন:



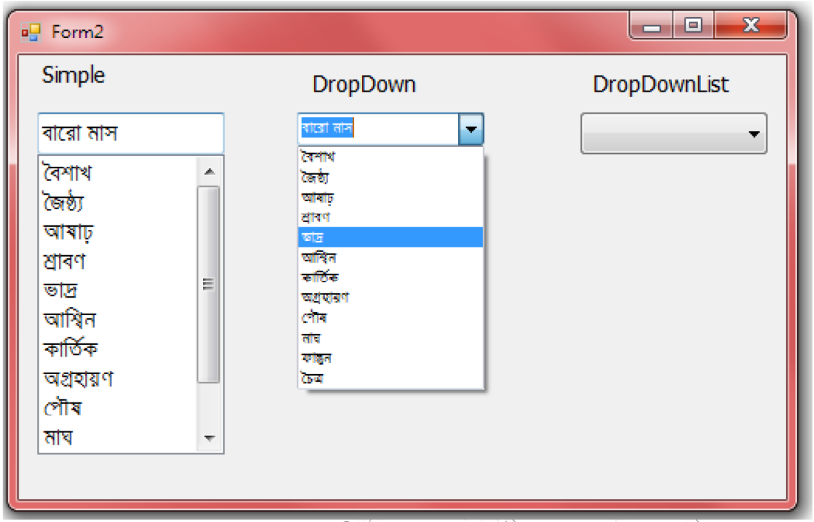

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    If ColorDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
        Button2.BackColor = ColorDialog1.Color
    End If
End Sub
```

উপরের চিত্রে রানকালীন অবস্থা তুলে ধরেছি। ডানের ডায়লগটিই হলো **ColorDialog**। আপনি যে কোনো রঙে ক্লিক করে OK দিন।

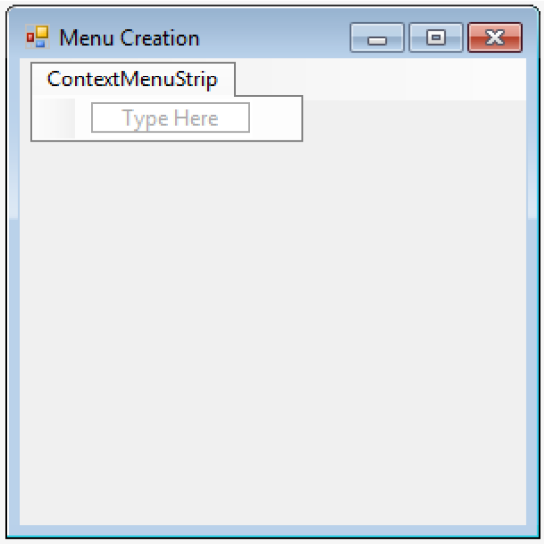
১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

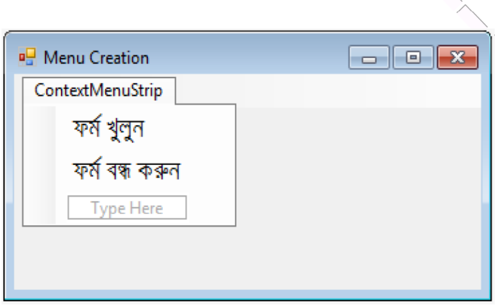
9. ComboBox: এটার ব্যবহার আমরা দেখেছি। তিন ধরনের কম্বোবক্স আছে: Simple, DropDown and DropDownList।



বায়ের চিত্রে তিন ধরনের কম্বোবাক্স দেখা যাচ্ছে। লক্ষণীয় যে, টুলবাক্সে কম্বোবাক্স একটাই। কিন্তু প্রোপার্টিতে এর ধরন নির্ধারণ করা যায়। এই প্রোপার্টির নাম DropDownStyle। এখানেই তিনটির যে কোনো একটি চয়েজ করতে পারেন।



10. ContextMenuStrip: এটা মাউসে রাইট-ক্লিক [Right-Click] করে মেনু তৈরির অবজেক্ট। ফর্মে নিলেই এটা মেনু তৈরির [ডিজাইনের] জন্য প্রস্তুতি নেবে। পাশের চিত্রটি দেখুন। এরপর টাইপ করে আপনার মেনু আইটেম যোগ করতে পারেন। লক্ষ করুন, আমাদের উদাহরণে বাংলায় মেনু আইটেম লিখেছি। ইউনিকোডে লিখলে কোনো সমস্যা নেই। এ উভয় মেনু আইটেমের ক্লিক ইভেন্ট হ্যান্ডলারে কোড লিখে আমরা নির্দেশ দিতে পারি। আমাদের উদ্দেশ্য ব্যবহারকারী প্রথম মেনু আইটেম চয়েজ করলে একটি ফর্ম খুলবে। দ্বিতীয়টি দ্বারা তা বন্ধ হয়ে যাবে। প্রথমে আমাদেরকে অবশ্য সেই ফর্মটি প্রজেক্টে যুক্ত করতে হবে। সুতরাং প্রজেক্ট মেনু থেকে একটি নতুন ফর্ম যুক্ত করুন। এবার এই কোডগুলো প্রথম ফর্মের ক্লাসের ভেতর লিখুন। মনে রাখবেন ফর্মের সাথে এই মেনুকে জড়িত করতে হবে। এটা করতে ফর্মে ক্লিক করুন। প্রোপার্টিতে যেয়ে ContextMenuStrip এর ডানে (none) শব্দের উপর ক্লিক করলেই দেখতে পাবেন ContextMenuStrip1। এটি চয়েজ করুন।

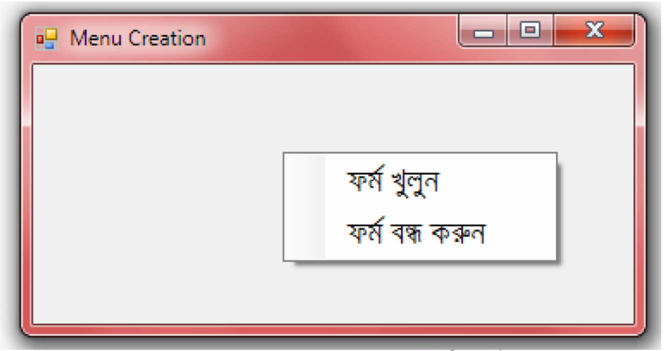


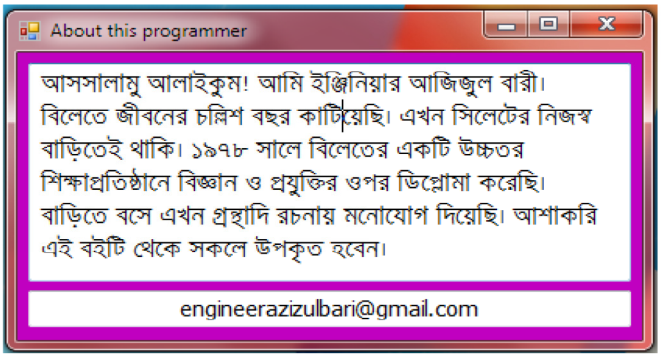

```
Private Sub ফর্মখুলুন_Click(sender As Object, e As EventArgs) Handles ফর্মখুলুন.Click
    If Form2.Visible = False Then
        Form2.Show()
        End If
    End Sub
Private Sub ফর্মবন্ধকরুন_Click(sender As Object, e As EventArgs) Handles ফর্মবন্ধকরুন.Click
    If Form2.Visible = True Then
        Form2.Close()
    End If
End Sub
```

প্রোগ্রাম রান করুন: পরের পৃষ্ঠার প্রথম [বামের] চিত্রে রানকালীন অবস্থায় প্রথম ফর্মে রাইট-ক্লিক করার পর কনটেক্স মেনু আসতে দেখাচ্ছে। ডানের চিত্রে কনটেক্স মেনু আইটেম থেকে ফর্ম খুলার লাইনটি চয়েজ করায় ফর্মটি খুলেছে।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

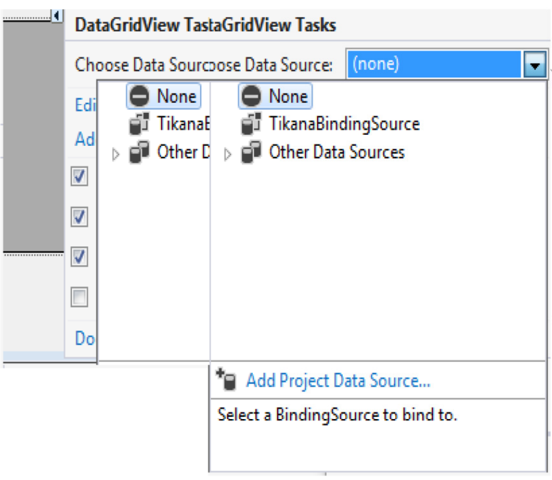
উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

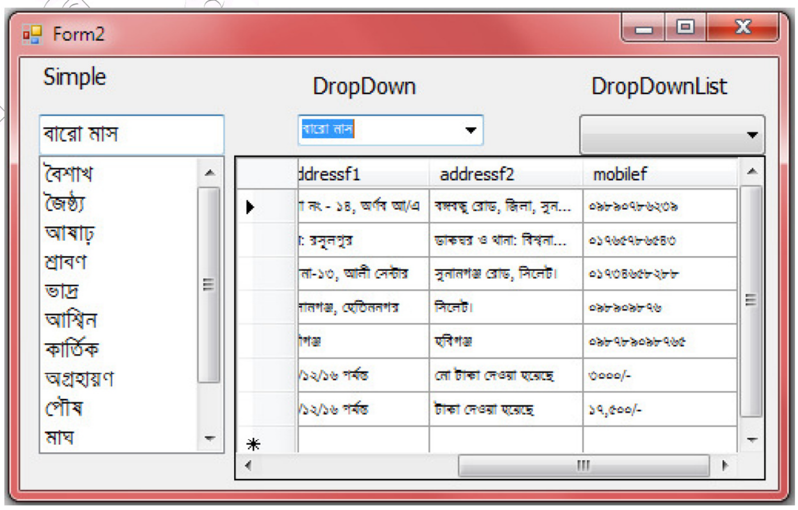




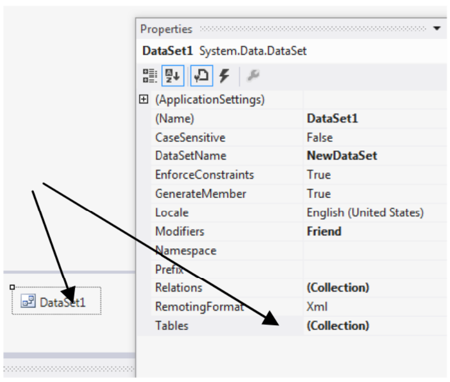
11. DataGridView: এটা ডাটাবেইজ প্রদর্শনের গ্রিড যা দেখতে অনেকটাই এক্সেস ডাটাবেইজের টেবিলের মতো। একটি উদাহরণ দেখে নিই।

মনে আছে কি- আমাদের ঠিকানা ডাটাবেইজের কথা? গ্রন্থের ১৪০ পৃষ্ঠায় চিত্রসহ নির্দেশনা দেওয়া আছে কিভাবে ডাটাবেইজের সঙ্গে যোগাযোগ সৃষ্টি করা যায়। এখানে অতিরিক্ত যা তাহলো, ডাটাগ্রিডভিউ ফর্মে নিয়ে আসার সাথে সাথেই একটি জানালা আসবে (নিচের বায়ের চিত্র)। যে কম্বোতে লিখা Add Project Data Source... আছে সেখানে যেয়ে নতুন ডাটাবেইজ আনার জন্য ক্লিক করুন। এরপর ১৪০ পৃষ্ঠায় যেভাবে নির্দেশিত হয়েছে সেভাবে ডাটাবেইজের সঙ্গে সংযোগ সৃষ্টি করে প্রোগ্রাম রান করুন [নিচের ডানের বায়ের চিত্র দ্রঃ]।





12. DataSet: এটা হলো ডাটাবেইজ 'মেমোরির মধ্যে উপস্থাপন'। এতে থাকতে পারে ডাটাটেবিল, ডাটারো ও ডাটাকলাম ইত্যাদি। ডাটাসেট ফর্মে নিয়ে আসলে তা নিচের ধূসর এলাকায় দেখা যাবে। এর প্রোপার্টি জানালায় যেয়ে বিভিন্ন ধরনের সম্পর্ক সৃষ্টি করতে পারেন। যেমন আপনি যদি চান ডাটাসেট মেমোরিতে একটি এক্সেস টেবিল রাখবেন, তাহলে আপনি এর প্রোপার্টিতে যেয়ে Tables -এ ক্লিক করলে ডানে কালেকশন শব্দ ও একটি বোতাম দেখতে পাবেন। এতে ক্লিক করলেই টেবিলের সঙ্গে যোগাযোগ তৈরি করতে পারেন।

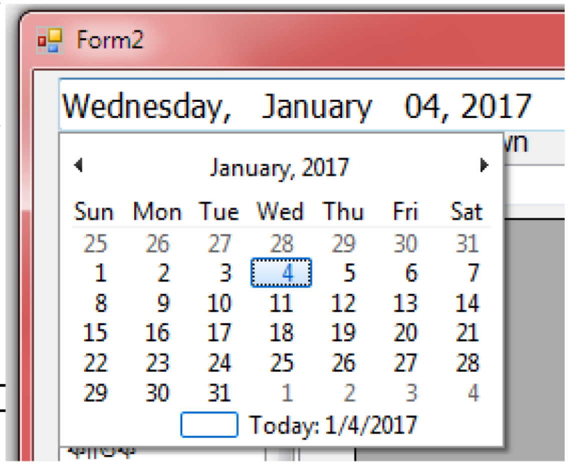


13. DateTimePicker: এই কন্ট্রোল দিয়ে তারিখ ও সময় প্রদর্শন করা যায়। কন্ট্রোলের প্রোপার্টিতে যেয়ে যেবাবে চান সেভাবে প্রদর্শনও সেট করতে পারেন। পরের পৃষ্ঠায় একটি চিত্রে আমরা এর ব্যবহার দেখিয়েছি। একটি ফর্মে এই কন্ট্রোল যুক্ত করে তার বিভিন্ন প্রোপার্টি কিভাবে কাজ করে তা দেখে নিতে পারেন।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

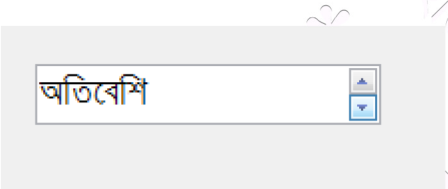
ডানের চিত্রে ডেইটটাইমপিকার দ্বারা ‘আজকের তারিখ’ একটি ক্যালেন্ডারে কিভাবে দেখানো হয় তা তুলে ধরেছি।



14. DirectoryEntry: এটার কাজ হলো ডাইরেক্টরির ভেতর কি হচ্ছে তা জানা। কোডের মাধ্যমে একে ব্যবহার করে যায়।

15. DirectorySearcher: এটার কাজ হলো ডমেইন সার্ভিস হাইআর্কিতে অনুসন্ধান চালানো।

16. DomainUpDown: এটা অনেকটা **CheckedListBox** এর মতো কাজ করে। এই কন্ট্রোলের ভেতর একটি লিস্ট থাকতে পারে। ডানের আপ-ডাউন এ্যারোতে ক্লিক করে করে লিস্ট দেখা যায়। নিচের চিত্রটি দেখুন।

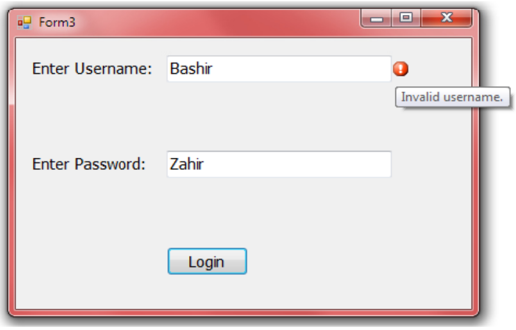


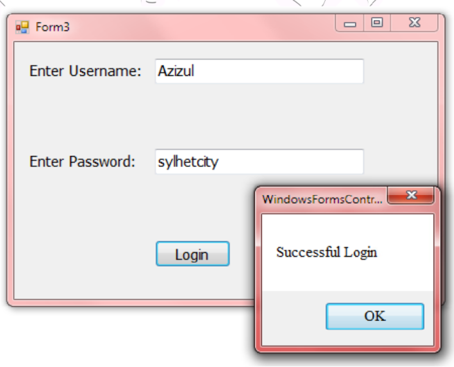
অবশ্য কোডের মাধ্যমে জানা ও সেট করা যায় লিস্টের কোন্ আইটেম চয়েজ করা হয়েছে বা হবে। যেমন:

DomainUpDown1.SelectedIndex = 0

17. ErrorProvider: এটা একটি সুবিধা। অর্থাৎ কোনো আইটেম ব্যবহারকারী প্রবেশ করলেন, আর তা আপনি যেভাবে চান তা হয় নি। এটা জানানোর জন্য আইকন ও মেসেজ ব্যবহারকারীকে দেখাতে পারেন। একটি উদাহরণ দ্বারা ব্যাপারটি স্পষ্ট হবে।

মনে করুন আপনি একটি লগিন ফর্ম বানাবেন। ব্যবহারকারী দু’টি ফিল্ডে ইউজারনেইম ও পাসওয়ার্ড প্রবেশ করবেন। আপনি একটি **ErrorProvider** দ্বারা লগিনের ভেলিডিটি চেক করতে পারেন। দেখুন নিচের চিত্র দু’টো ও পরের কোডগুলো।






```
Private Sub btLogin_Click(sender As Object, e As EventArgs) Handles btLogin.Click
    If TextBox1.Text <> "Azizul" Then
        ErrorProvider1.SetError(TextBox1, "Invalid username.")
        Exit Sub
    Else
        ErrorProvider1.SetError(TextBox1, "")
    End If
    If TextBox2.Text <> "sylhetcity" Then
        ErrorProvider1.SetError(TextBox2, "Invalid password.")
        Exit Sub
    Else
        ErrorProvider1.SetError(TextBox2, "")
    End If
    MsgBox("Successful Login")
End Sub
```

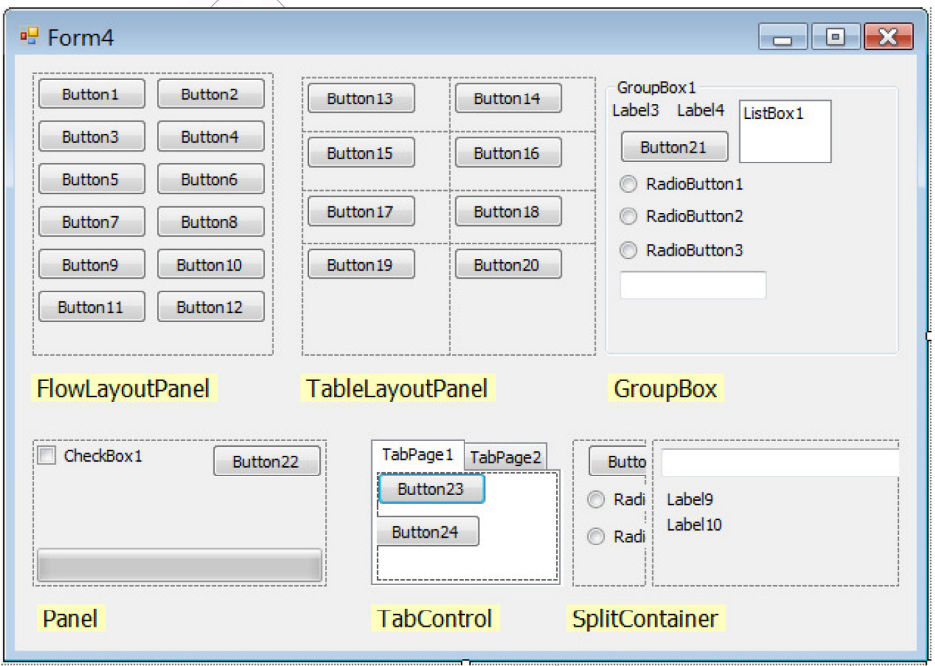
১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

18. EventLog: এটার মাধ্যমে কোনো এ্যাপলিকেশনের ইভেন্ট লগ [ঘটনাপ্রবাহ রেকর্ড] তৈরি করা যায়। এটা আমরা খুব অল্প ক্ষেত্রেই ব্যবহার করে থাকি। সুতরাং এ সম্পর্কে বেশি বলার নেই।

19. FileSystemWatcher: এটার কাজ হলো ফাইল সিস্টেমে কী রদবদল হচ্ছে তা জানা ও প্রয়োজনে রিপোর্ট করা। এটা একটি এডভান্স স্তরের ক্রিয়া তাই এটা ব্যবহার পদ্ধতি এখানে দেখানোর প্রয়োজন নেই।

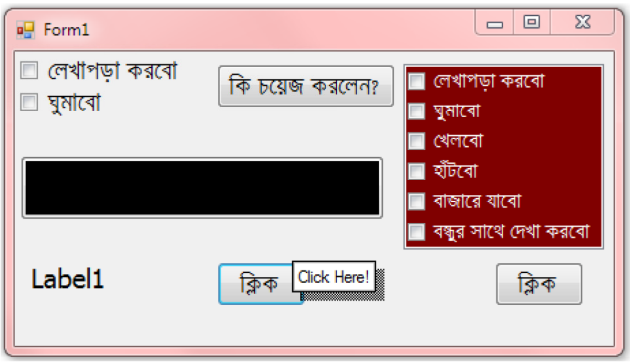
20. FlowLayoutPanel: -এটা Panel জাতীয় কয়েকটি কন্ট্রোলের একটি। অন্যান্যগুলো হলো: TableLayoutPanel, GroupBox, TabControl, SplitContainer এবং Panel। এগুলো মূলত ডিজাইনকালে সুবিধার জন্য ব্যবহৃত হয়। নিচের চিত্রটি দেখুন।



22. HelpProvider: যে কোন কন্ট্রোলের জন্য সাহায্য নিশ্চিত করার উপায় হিসাবে এই কন্ট্রোলটি ব্যবহৃত হয়ে থাকে। যেমন মনে করুন, আপনি চাচ্ছেন একটি বোতামের একশন কি হবে তা ব্যবহারকারীকে জানাবেন। আপনি চাইলে টুলটিপ ব্যবহার করতে পারেন। তবে এরচেয়েও উন্নতমানের পথ হলো হেল্পপ্রভাইডার। এটির মাধ্যমে শুধুমাত্র সামান্য কিছু লেখা নয়- আরো একটি সংরক্ষিত হেল্প ফাইলও দেখাতে পারেন। ব্যবহারকারী কন্ট্রোলটি ফোকাসে আসার পর কিবোর্ডের ‘’ -এ প্রেস করলেই হেল্পপ্রভাইডার সক্রিয় হয়ে ওঠবে। আসুন, একটি উদাহরণ দেখে নিই- বিষয়টি এতে স্পষ্ট হয়ে যাবে। আমরা ফর্মের লোড ইভেন্ট সাবে এই কোডটুকু লিখবো:

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    HelpProvider1.SetHelpString(Button3, "Click Here!")
End Sub
```

অবশ্য ফর্মের গ্রে এলাকায় একটি `HelpProvider1` আইটেম তো আনতে হবে। টটুলবাক্সে যেয়ে এই আইটেমে ডবলক্লিক করলেই কাজটি সারবে। রান করার পর বোতামটি ফোকাসে এলে ‘F1’ কী চাপুন। ডানের চিত্র দেখুন।



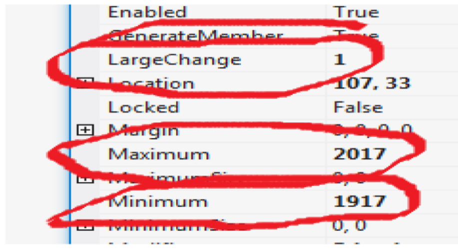
১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

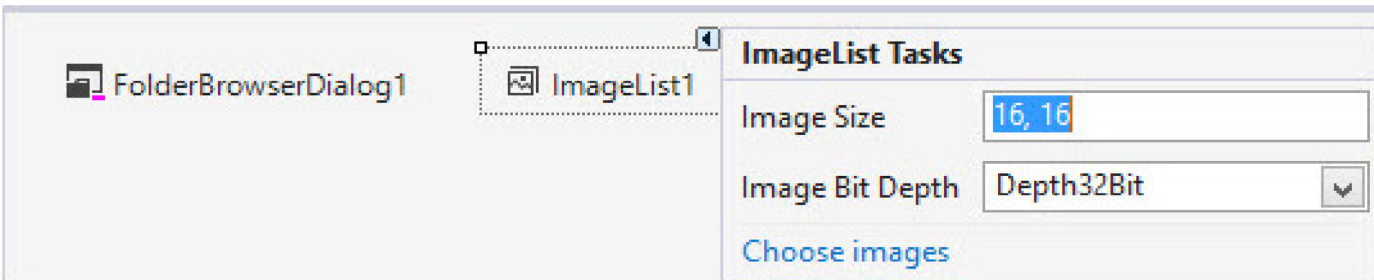
23. HScrollBar: এটা ও 'VScrollBar' কন্ট্রোলের কাজ হলো যথাক্রমে ব্যবহারকারী একটি অংশে ধরে অনুভূমিক ও উপর-নিচে স্ক্রল করতে পারবেন। আমরা একটি উদাহরণ দ্বারা এগুলো বুঝিয়ে দিচ্ছি।



বায়ের চিত্রে আমরা দু'টি স্ক্রল আইটেম দেখছি। এগুলোর মাধ্যমে লেবেল দু'টোর সংখ্যা রদবদল করা যায়। আমরা ডিজাইন টাইমে সর্বোচ্চ ও সর্বনিম্ন সংখ্যা নির্ধারণ করে দিতে পারি। কোন্ কোন্ প্রপার্টি পরিবর্তন করে তা সম্ভব নিচের চিত্রে দেখিয়েছি।



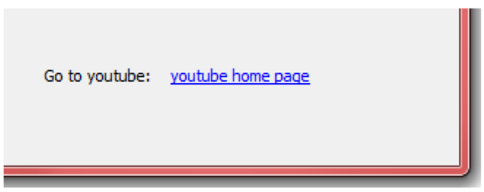
24. ImageList: এটা ছবির একটি লিস্ট স্টোর করে যাতেকরে অন্যান্য কন্ট্রোল বা কোড এগুলো কাজে লাগাতে পারে। আপনি প্রথমে ফর্মে একটি ইমেজলিস্ট নিয়ে আসবেন টুলবক্স থেকে। দেখবেন, এর ডানে একটি ছোট্ট এ্যারো আছে। সিলেক্ট করলেই ইমেজলিস্ট ট্যাস্কস লিস্ট আসবে। আপনি সেখানে ইমেজ সম্পর্কে তথ্যাদি লিখে রাখতে পারেন। ইমেজ লিস্ট নিয়েও আসতে পারেন। দেখুন নিচের চিত্রটি।



25. Label: এটা সম্পর্কে আর কী বলবো। আমরা তো সব সময় এটা ব্যবহার করে আসছি। ব্যবহারকারীকে দৃশ্যত কিছু তথ্য প্রদানের জন্য এটি কাজে লাগানো হয়। অবশ্য রানটাইমে কোডোর মাধ্যমে এর টেক্সট, ভিজিবল ইত্যাদি প্রোপার্টি পরিবর্তনও করা যায়।

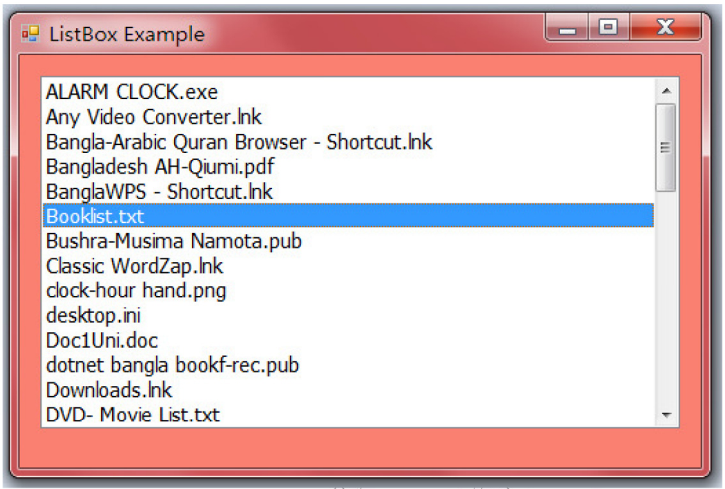
26. LinkLabel: এটাও মূলত একটি লেবেল। তবে ওয়েভ পেইজে ন্যাভিগেশনের জন্য এটি সহজে ব্যবহার করা যায়। বা ফর্মের ভেতরেও অন্যান্য কাজের জন্য এটি ব্যবহারযেগ্য। এর লেখা থাকে নীল রঙের। এছাড়া আন্ডারলাইনও থাকে। এর উপরে মাউস গেলে হাত আইকনে পরিণত হয়। দেখুন নীচের চিত্রটি। লিংকলেবেলের বেশ কয়েকটি প্রোপার্টি আছে। এগুলো রদবদল করে দেখতে পারেন। একটি ওয়েভসাইটে সরাসরি ভিজিটের জন্য ফর্মের লিংকলেবেলে নিচের কোডটুকু লিখুন।

```
Private Sub LinkLabel1_LinkClicked(sender As Object, e As LinkLabelLinkClickedEventArgs) Handles
LinkLabel1.LinkClicked
        ' open the website URL set to the Tag property
        ' (www.youtube.com)
        Process.Start(LinkLabel1.Tag.ToString)
End Sub
```



১৭তম পরিচ্ছেদ: *Windows Form Controls*

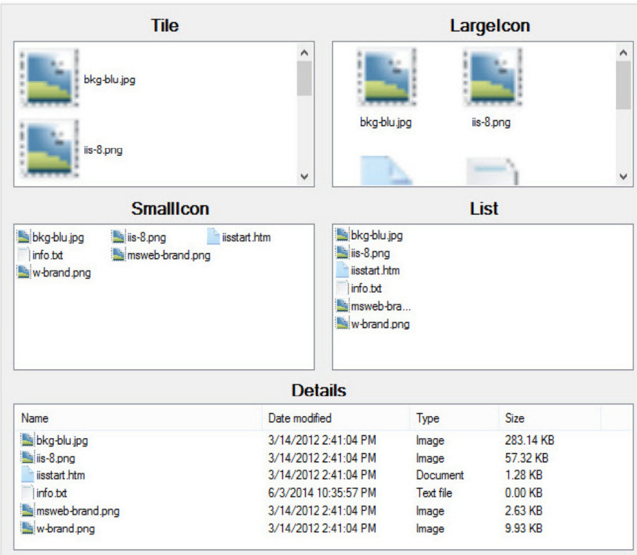
উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*



27. ListBox: এটাও আমরা ইতোমধ্যে ব্যবহার করেছি। এতে থাকে নিয়ন্ত্রণযোগ্য কিছু লিস্ট আইটেম। লিস্ট হবে ০ ইনডেক্স। অর্থাৎ চারটি আইটেম থাকলে ইনডেক্স হবে ০,১,২ ও ৩। ইনডেক্স দ্বারা, কিংবা লাইনের সিলেকশন টেক্সট দ্বারা লিস্ট বাক্সের আইটেম সনাক্ত করা যায়। আমরা একটি উদাহরণ দেখাবো। বায়ের চিত্রে একটি জানালার মধ্যে আমরা ফাইলের একটি লিস্ট দেখছি। বাস্তবে এই জানালায় একটি লিস্ট বাক্স আছে। কোডিং এর মাধ্যমে আমরা কম্পিউটারের ডেস্কটপ ফল্ডারের সকল ফাইলের নাম এই বাক্সে প্রদর্শন করেছি। নিচে এই কোডটুকু দেখুন। একটি ফর্ম আপনার প্রজেক্টে যুক্ত করুন। এতে টুলবাক্স থেকে নিয়ে আসুন একটি লিস্টবক্স। ফর্মের কোথাও ডবল ক্লিক করুন। কোড এডিটরের Form_Load ইভেন্ট সাবে নিচের কোডটি লিখুন। আপনি চাইলে অন্য যে কোনো ফল্ডারের নামও দিতে পারেন। বরং বিভিন্ন ফল্ডানের নাম দিয়ে টেস্ট করে দেখুন।

```
Private Sub Form6_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    Dim mydoc As String = My.Computer.FileSystem.SpecialDirectories.Desktop
    Dim path As String = IO.Path.Combine(mydoc)
    Dim dir As New IO.DirectoryInfo(path)
    ' populate the listbox control
    For Each file In dir.GetFiles
        ' avoid jpg images
        If Not file.Extension = ".jpg" Then
            ListBox1.Items.Add(file.Name)
        End If
        Next
End Sub
```

28. ListView: এটাতে ফাইলের লিস্ট দেখানো যায়। তবে আরো বেশি তথ্যসহ। মোট ৫ উপায়ে লিস্টভিউ কন্ট্রোলে ফাইল নাম প্রদর্শন সম্ভব। বায়ের চিত্রটি দেখুন।



ক. Tiles: আইটেমের নামের বামপাশে বড়ো বাড়ো আইকন দেখায়।

খ. LargeIcon: আইটেমের নামের উপরে বড়ো আইকন দেখায়।

গ. SmallIcon: নামের বায়ে ছোট্ট আইকন প্রদর্শন করে।

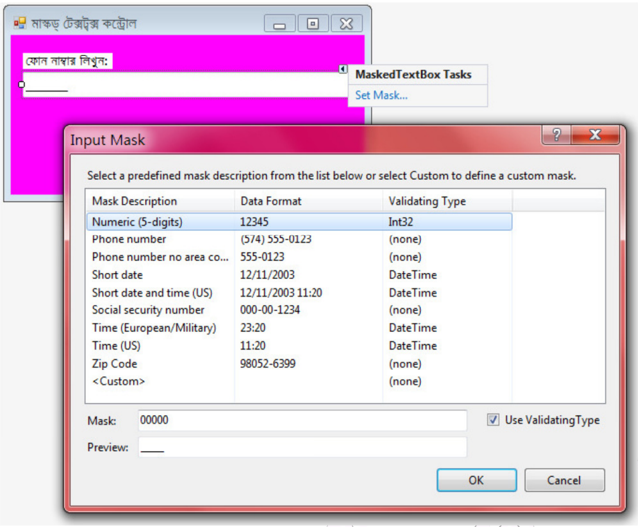
ঘ. List: নামের বায়ে ছোট্ট আইকন দেখায় ও প্রতিটি আইটেম একেকটি লাইনে থাকে।

ঙ. Details: ছোট্ট আইকনসহ ফাইলের পূর্ণাঙ্গ তথ্য প্রদর্শন করে।

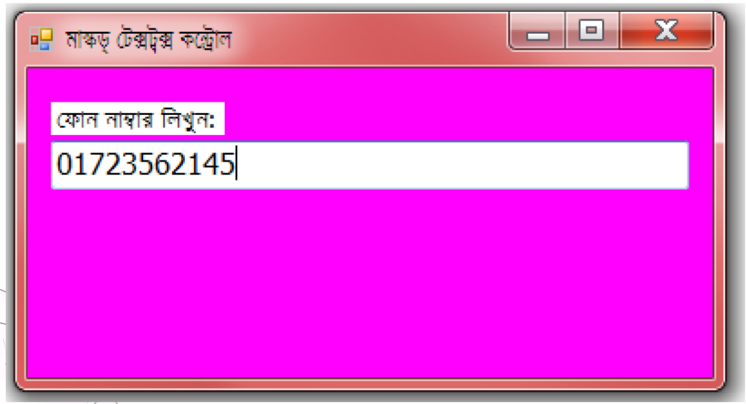
29. MaskedTextBox: এটা টেক্সট বাক্সের মতোই কাজ করে। তবে অতিরিক্ত যা তাহলো ব্যবহারকারী ফিল্ডের মধ্যে কী লিখবেন তা নির্ধারণ করা যায়। এক ধরনের ভেলিডিটি ফিল্ড এটি। উদাহরণ দেখুন।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

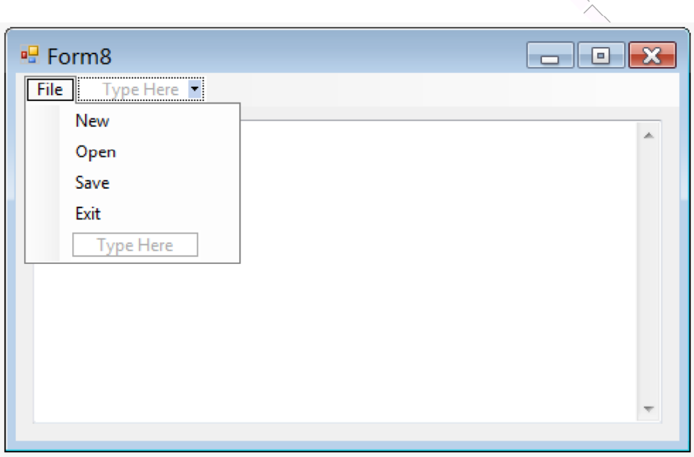


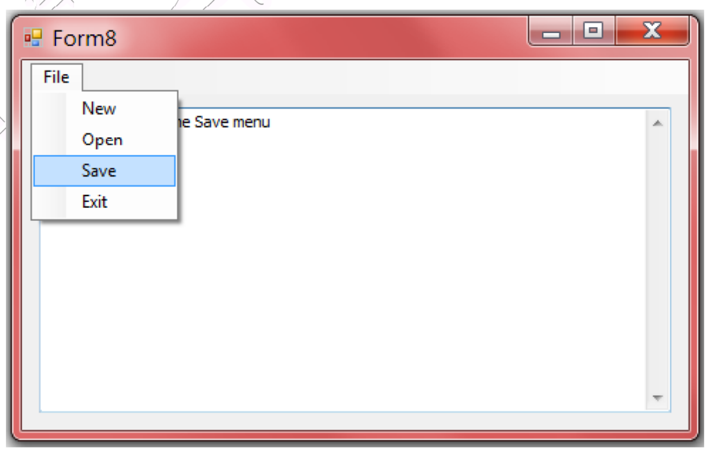
টুল বাক্স থেকে ফর্মে যুক্ত করুন **MaskedTextBox** একটি । টেক্সবাক্সের ডানের দিকে উপরের কোণে যে এ্যারো দেখা যাবে তাতে ক্লিক করুন। চয়েজ করুন Set Mask …। বাঁয়ে চিত্রিত জানালাটি আসবে। এখান থেকে যে কোনো একটি চয়েজ করে ‘OK ক্লিক করুন। এবার প্রোগ্রাম রান করুন। দেখবেন আপনার



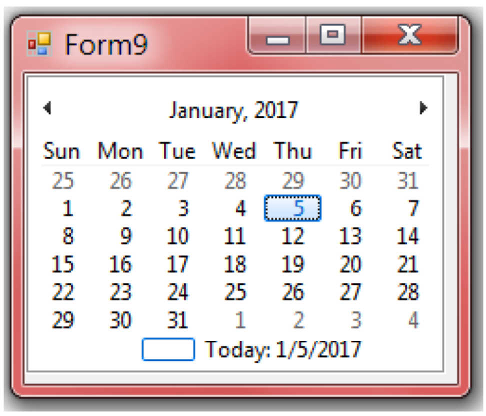
চয়েজকৃত উপায়েই লেখা ঢুকানো সম্ভব হচ্ছে টেক্সট ফিল্ডে। লক্ষ করুন, ফিল্ডের মধ্যে ১১টি মাত্র সংখ্যা ঢুকানো সম্ভব হয়েছে।

30. MenuStrip: এ কন্ট্রোলের মাধ্যমে ফর্মের মধ্যে মেনু, সাবমেনু, মেনু আইটেম ইত্যাদি তৈরি করা যায়। প্রত্যেকটির সঙ্গে কোড যুক্ত করে কাঙ্ক্ষিত একশনও আঞ্জাম দেওয়া যায়। নিচের চিত্রগুলো দেখুন।





31. MessageQueue এটা একটি উচ্চ পর্যায়ের ক্লাস। সার্ভারের সাথে মেসেজ যোগাযোগ কিউ নিয়ন্ত্রণে এটি ব্যবহৃত হয়। এ টুলবক্স আইটেম সম্পর্কে আর অতিরিক্ত আলোচনা গ্রন্থের স্কোপের বাইরে। সুতরাং আর কিছু বলার নেই। এ সম্পর্কে অতিরিক্ত জানতে চাইলে ইন্টারনেটের এই লিংক দেখতে পারেন: http://msdn.microsoft.com/en-us/library/system.messaging.messagequeue.aspx



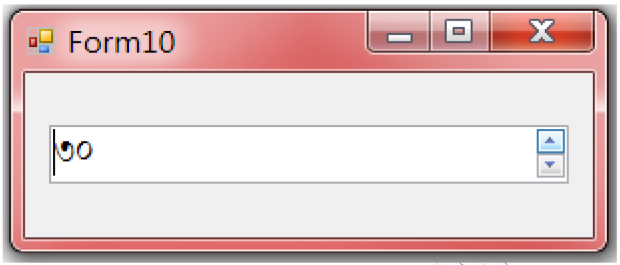
32. MonthCalender বুঝতেই পারছেন, এটা হলো পুরো মাসের ক্যালেন্ডার কন্ট্রোল। ক্যালেন্ডারের রেঞ্জ বিরাট: ১৭৫৩ থেকে ৯৯৯৮ সাল পর্যন্ত এ রেঞ্জের ব্যাপ্তি। আপনি এরই ভেতর যে কোনে তারিখ যথাক্রমে সর্বনিম্ন ও সর্বোচ্চ হিসেবে প্রোপার্টিতে যেয়ে সেট করতে পারেন। ডানের চিত্রটি দেখুন।

33. NotifyIcon: এটা প্রোগ্রাম রানকালে অদৃশ্য থাকে। একে ব্যবহার করা

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

যায় সিস্টেম ট্রেতে আইকোন প্রদর্শন করে।



34. **NumericUpDown:** এ কন্ট্রোল একটি সংখ্যা প্রদর্শন করে। এতে থাকে একটি আপ-ডাউন এ্যারো। আপনি প্রোপার্টিতে যেয়ে সর্বোচ্চ ও সর্বনিম্ন সংখ্যা, কত করে বাড়বে বা কমবে ক'টি ডেসিলেম স্থান হবে ইত্যাদি সেট করতে পারেন। কন্ট্রোলের ValueChanged ইভেন্ট সক্রিয় হয় যখনই ব্যবহারকারী কিংবা কোড এর মান পরিবর্তন করে। এই ইভেন্ট সাবকে ধরে কোডিং করা যায়। দেখুন বায়ের চিত্রটি।

35. OpenFileDialog -এটার মাধ্যমে ফাইল খোলার ডায়লগ প্রদর্শিত হয়। এটা একটি শক্তিশালী কন্ট্রোল। অবশ্য ভিজুয়্যাল বেসিক.নেট বেশ ক'টি অনুরূপ ডায়লগ সৃষ্টি করে দিয়েছে। অপেন ডায়লগের কাজ হলো কম্পিউটার থেকে কোনো ফাইল সিলেক্ট করে খোলার জন্য প্রস্তুত করা। ডায়লগের ফলাফল হয় '**OK**' না হয় '**Cancel**'। এই কন্ট্রোলে বেশ ক'টি প্রোপার্টি আছে। এগুলো রানটাইমে ব্যবহার করা যায়। যেমন Filter প্রোপার্টির মাধ্যমে নিশ্চিত করতে পারেন যে ব্যবহারকারী শুধুমাত্র .jpg ও .png ফাইল টাইপ খোলা ছাড়া অন্য কিছু দেখতেই পারেন না। আমরা কোডে লিখতে পারি:

```
OpenFileDialog1.Filter = "Jpeg Files (*.jpg)|*.jpg|Png Files (*.png)|*.png"
```

যে ফাইল সিলেক্ট করা হয়েছে তার নাম জানতে হলে আমরা ব্যবহার করতে পারি FileName প্রোপার্টি, যেমন:

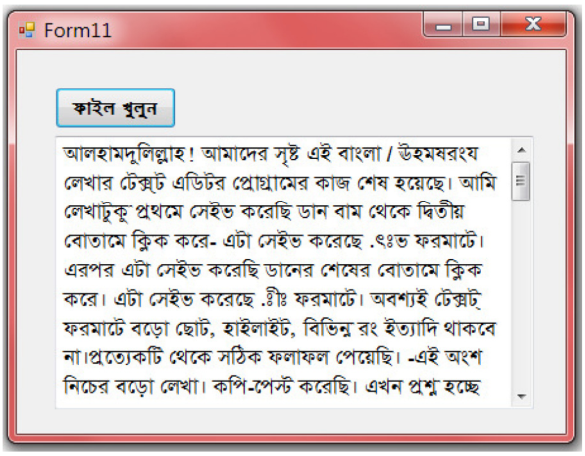
```
Private Sub BrowseButton_Click(sender As Object, e As EventArgs) Handles BrowseButton.Click
    If OpenFileDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
        SelectedFile.Text = OpenFileDialog1.FileName
    End If
End Sub
```

আরো একটি উল্লেখযোগ্য প্রোপার্টি হচ্ছে MultiSelect । এটা দ্বারা সিদ্ধান্ত নেওয়া যায় ব্যবহারকরী একাধিক ফাইল সিলেক্ট করতে পারবে কি না। যেমন:

```
Private Sub BrowseButton_Click(sender As Object, e As EventArgs) Handles BrowseButton.Click
    If OpenFileDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
        For Each file As String In OpenFileDialog1.FileNames
            MessageBox.Show(file)
        Next
    End If
End Sub
```

মেসেজ বাক্সে একে একে সকল ফাইলের নাম প্রদর্শিত হবে। এবার আমরা ছোট্ট একটি উদাহরণ দেখে নেবো।

একটি নতুন ফর্ম প্রজেক্ট তৈরি করুন। যোগ করুন একটি বোতাম, একটি টেক্সট বাক্স ও একটি অপেন ফাইল ডায়লগ কন্ট্রোল। বোতামে ডবল ক্লিক করুন। কোড জানালাতে ফর্মের ক্লাসের ভেতর যে ইভেন্ট সাব আসবে তার ভেতর লিখুন:

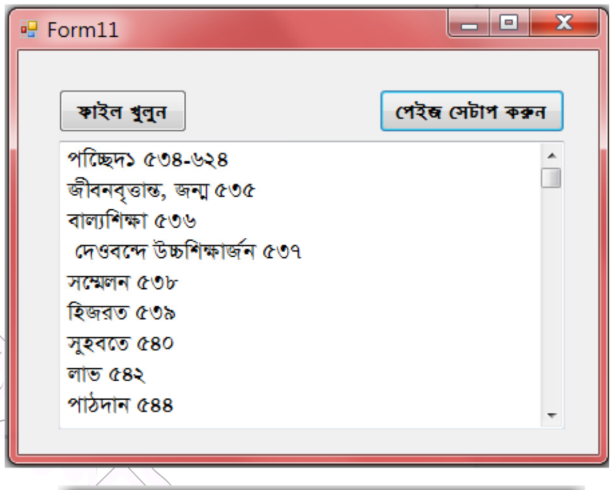



```
Public Class Form11
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button1.Click
        ' Displays an OpenFileDialog so the user can select a Cursor.
        Dim openFileDialog1 As New OpenFileDialog()
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

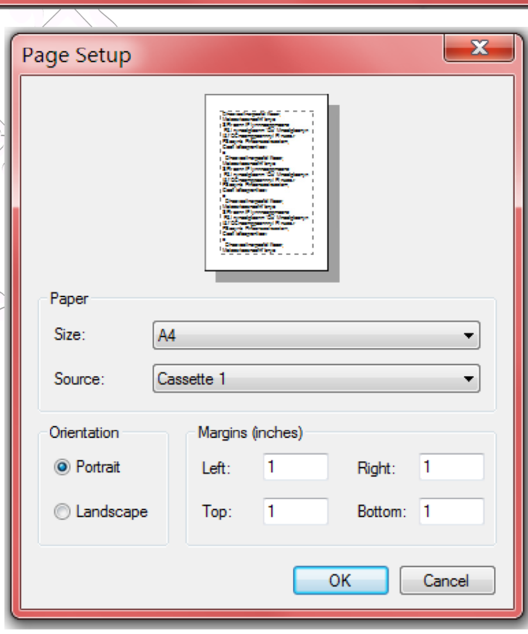
উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
  openFileDialog1.Filter = "Text Files|*.txt"
    openFileDialog1.Title = "Select a Text File"
    If openFileDialog1.ShowDialog() = System.Windows.Forms.DialogResult.OK Then
       Dim sr As New System.IO.StreamReader _(openFileDialog1.FileName)
       TextBox1.Text = (sr.ReadToEnd)
       sr.Close()
    End If
  End Sub
End Class
```



36. PageSetupDialog: এই ডায়লগটি দ্বারা পেইজ সেটাপ করা যায়। একটি উদাহরণ দেখুন। আমরা ইতোমধ্যে অপেনডায়লগ নিয়ে যে প্রোগ্রামটি করেছিলাম সেটি আবার খুলুন। ফর্মে যোগ করুন একটি পেইজসেটাপডায়লগ ও একটি ‘পেইজ সেটাপ করুন’ বোতাম। এ বোতামের ইভেন্ট সাবরুটিনে লিখুন:

```
Private Sub Button2_Click(sender As Object, e As EventArgs)
Handles Button2.Click
    ' Initialize the dialog's PrinterSettings property to hold user
    ' defined printer settings.
    PageSetupDialog1.PageSettings = _
       New System.Drawing.Printing.PageSettings
    ' Initialize dialog's PrinterSettings property to hold user
    ' set printer settings.
    PageSetupDialog1.PrinterSettings = _
       New System.Drawing.Printing.PrinterSettings
    'Do not show the network in the printer dialog.
    PageSetupDialog1.ShowNetwork = False
    'Show the dialog storing the result.
    Dim result As DialogResult = PageSetupDialog1.ShowDialog()
    ' If the result is OK, display selected settings in
    ' ListBox1. These values can be used when printing the
    ' document.
    If (result = DialogResult.OK) Then
       Dim results() As Object = New Object() _
       {PageSetupDialog1.PageSettings.Margins, _
       PageSetupDialog1.PageSettings.PaperSize, _
       PageSetupDialog1.PageSettings.Landscape, _
       PageSetupDialog1.PrinterSettings.PrinterName, _
       PageSetupDialog1.PrinterSettings.PrintRange}
    End If
End Sub
```



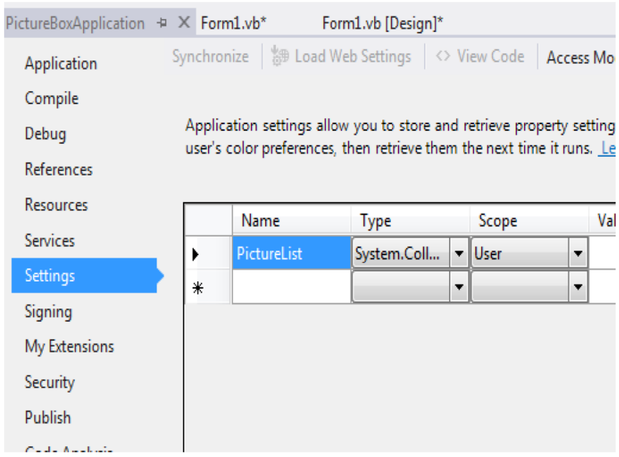
37. Panel: প্যানেল কন্ট্রোল মূলত গ্রুপ বাক্সের মতো। আমরা ১৭৭ পৃষ্ঠায় এর ব্যবহার দেখেছি। সুতরাং এখানে আর পুনরাবৃত্তি করছি না।

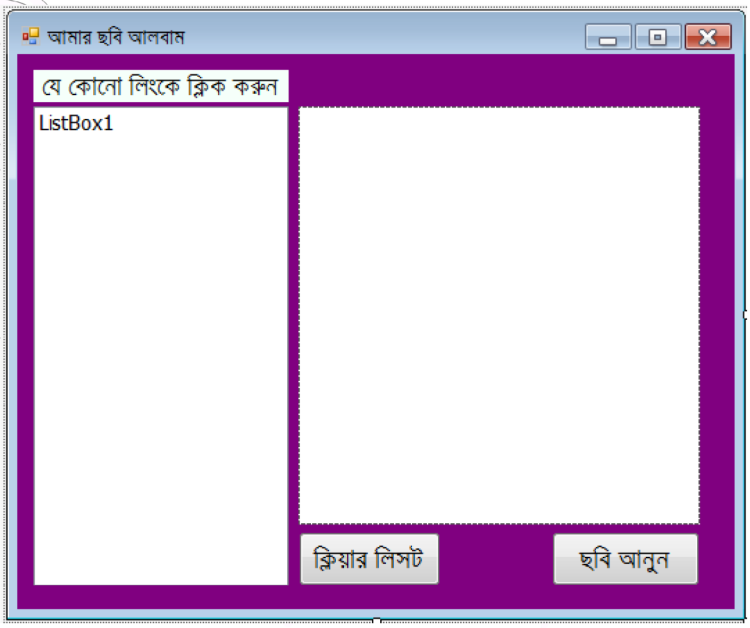
38. PerformanceCounter: এটা হলো উইন্ডোজ এনটি [Windows NT] পারফরম্যান্স কাউন্টার কম্পোনেন্ট। এটা একটি উচ্চ পর্যায়ের কন্ট্রোল। কম্পিউটারের সিস্টেমের সঙ্গে এর ইন্টারেকশন। বর্তমান গ্রন্থে এর ওপর আদৌ কোনো অতিরিক্ত আলোচনা ঠিক হবে না।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

39. PictureBox: এই কন্ট্রোলের মাধ্যমে ছবি প্রদর্শন করা যায়। এছাড়া এটি একটি গ্রাফিক্স অবজেক্ট হিসেবেও কাজ করে যার মধ্যে আপনি বিভিন্ন ধরনের আকার রানটাইমে অঙ্কন করতে পারেন। ডিজাইন টাইমে দু'টি প্রোপার্টি খুব ব্যবহারোপযোগী। এগুলো হলো তার ইমেজ [Image] এবং সাইজমোড [SizeMode] প্রোপার্টি। প্রথমটি দ্বারা যে কোনো ছবি ডিফোল্ট হিসাবে প্রদর্শন ও দ্বিতীয়টি দ্বারা ছাবিটি কিভাবে প্রদর্শিত হবে তা নির্দিষ্ট করা যায়। ছবিটি কন্ট্রোলের পুরো এলাকা জুড়ে, মধ্যখানে, বড় হয়ে ইত্যাদি কোন্ উপায়ে প্রদর্শিত হবে- তা জিাইনটাইমে করার সুযোগ সৃষ্টি করে দিয়েছে দ্বিতীয় প্রোপার্টিটি। পিকচার বাক্স সম্পর্কে আরো কিছু জানতে আমরা একটি সুন্দর এ্যাপলিকেশন তৈরি করবো। এই এ্যাপলিকেশন হবে ছবি আলবামের মতো। আপনি যতোটা প্রয়োজন ছবি কম্পিউটারের যে কোনো ফল্ডার থেকে এনে পিকচার বাক্সে প্রদর্শন করতে পারেন। আরো যা পারবেন তাহলো পরবর্তিতে প্রোগ্রাম রান করলে সকল ছবির লিংক একটি লিস্ট বাক্সে দেখতে পাবেন। শুধু মাত্র যে কোনো লিংকে ক্লিক করলেই ছবিটি ভেসে আসবে পিকচার বাক্সে। সুতরাং আপনার ছবি আলবাম বানাতে নিচের পদক্ষেপগুলো অনুসরণ করুন।





১. একটি ফর্ম প্রজেক্ট তৈরি করুন। এর নামকরণ করুন: MyPictureAlbum।

২. ফর্মে টুলবাক্স থেকে নিয়ে আসুন দু'টি বোতাম, একটি পিকচার বাক্স, একটি লেবেল, একটি লিস্ট বাক্স ও একটি অপেন ফাইল ডায়লগ। এরপর ইচ্ছেমতো ডিজাইন করুন। দেখুন আমার ডিজাইনটি।

৩. Solution Explorer এ যেয়ে ডবল ক্লিক করুন, My Project। একটি জানালা আসবে। নিচে এর একটি অংশ দেখানো হলো।

৪. সিলেক্ট করুন, Settings। Name কলামে টাইপ করুন: PictureList। পাশের Type কলামের এ্যারোতে যেয়ে সিলেক্ট করুন: System.Collections.Specialized.StringCollection। ব্যস! এটি বন্ধ করুন। এই সেটিংয়ের কারণে আমরা লিস্ট বাক্সের আইটেমগুলো সংরক্ষণ করতে পারবো।

৫. এবার কিছু কোডিং লাগবে। (ক) 'ছবি আনুন' বোতামে ডবল ক্লিক করে কোড এডিটর খুলুন। বোতামের ক্লিক ইভেন্টে লিখুন:

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim result As DialogResult
    'set filter
    OpenFileDialog1.Filter = "JPEG Files (*.jpg)|*.jpg|Bitmap Files(*.bmp)|*.bmp| Gif files(*.gif)|*.gif| Png files
(*.png)|*.png"
    'Enable multi-file names selection
    OpenFileDialog1.Multiselect = True
    'Open file dialog
    result = OpenFileDialog1.ShowDialog
    If result = DialogResult.OK Then
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
        For Each filename In OpenFileDialog1.FileNames
            'add selected file names to listbox
            ListBox1.Items.Add(filename)
        Next
    End If
End Sub
```

এই কোডের কাজ হলো OpenFileDialog1 এর মাধ্যমে এক বা একাধিক ছবির পূর্ণাঙ্গ প্যাথ স্ট্রিং লিস্টবাক্সে নিয়ে আসা। প্রথম লাইন দ্বারা আমরা ডায়লগ থেকে যে ফলাফল আসবে তার জন্য একটি ডিক্লারেশন করেছি। দ্বিতীয় লাইন দ্বারা নির্দিষ্ট করেছি কোন্ কোন্ ধরনের ইমেজ ফাইল আনা যাবে। এর পরের লাইনে অপেন ডাইলগের Multiselect নামক প্রোপার্টি সত্য হিসাবে সেট করেছি। এর ফলে একই সঙ্গে একাধিক ছবি সিলেক্ট করতে পারবেন ব্যবহারাকারী। এর পরের লাইনে ব্যবহারকারী ক্যানসেল না করলে আমরা প্রতিটি সিলেক্টেট ছবির ফাইন নাম [তাদের প্যাথ নেইম] লিস্ট বাক্সে নিয়ে এসেছি। অবশ্য ইতোমধ্যে লিস্ট বাক্সে কিছু থাকলে তা মুছে যাবে না।

৬. লিস্ট বাক্সে ডবল ক্লিক করুন। যে সাব আসবে তাতে লিখুন এই কোডটুকু:

```
Private Sub ListBox1_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
    ListBox1.SelectedIndexChanged
    'View image in picturebox when a file name was selected from the list
    PictureBox1.Image = Image.FromFile(ListBox1.SelectedItem.ToString)
End Sub
```

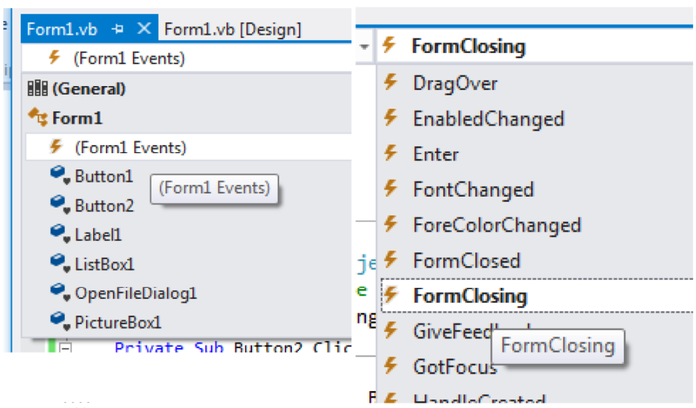
এখানো আমরা একটি মাত্র লাইন লিখেছি। লিস্ট বাক্সে যে ইনডেক্সে ক্লিক করা আছে সে প্যাথ নাম অনুযায়ী পিকচার বাক্সে ছবি এনে প্রদর্শনের নির্দেশ দেওয়া হয়েছে এ লাইনের মাধ্যমে।

৭. এবার ডবল ক্লিক করুন ক্লিয়ার লিস্ট বোতামে। ক্লিক ইভেন্ট সাবে লিখুন:

```
'clear list button code
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    PictureBox1.Image = Nothing
    Dim Res As Integer = MsgBox("সকল প্যাথ নাম মুছে দেবেন?", MsgBoxStyle.YesNo)
    ' MsgBox(Res)
    If Res = 6 Then
        ListBox1.Items.Clear()
    End If
End Sub
```

এই সাবের প্রথম লাইন দ্বারা প্রদর্শিত ইমেজ (পিকচার বাক্সে থাকলে) মুছে দিয়েছি। এরপর একটি ওয়ার্নিং দ্বারা ব্যবহাকরীকে জানানো হচ্ছে সত্যিই কি লিস্টটি মুছে দেবেন? ফলাফল সত্যি হলে [৬] পরের লাইন নির্দেশ দেওয়া হয়েছে লিস্ট বাক্সের সবকিছু মুছে ফেলে দিতে।

৮. Form Load এবং Form Closing ইভেন্ট সাবের কোড: আমরা এ দু'টো ইভেন্ট ধরে কিছু কাজ করেছি। লোড হওয়ার সময় আমরা চাই সিস্টেম সেটিংয়ে সংরক্ষিত লিস্টটি লিস্ট বাক্সে নিয়ে আসতে। আর বন্ধ হওয়ার সময় আমরা চাই লিস্ট বাক্সের আইটেমগুলো আবার সিস্টেম সেটিংয়ে সংরক্ষণ করতে- কারণ, ইতোমধ্যে ব্যবহারকারী লিস্টে পরিবর্তন আনতে পারেন। প্রথম সাবরুটিন আসবে ফর্মের টাইলেট বারে ডবল ক্লিক করলে। কিন্তু দ্বিতীয়টি ফর্মের ইভেন্ট লিস্ট থেকে আনতে হবে। দেখুন ডানের চিত্রটি। এটা আনার ক্লিক পরম্পরা: ক্লিক Form1.vb > Form1 > (Form1 Event > FormClosing। ব্যস! `Form1_FormClosing` নামক একটি ইভেন্ট হ্যান্ডলার আসবে।



১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

যাক, প্রথমে আমরা ফর্ম লোডের কোডের ওপর ব্যাখ্যা দেবো। নিচে এই কোডগুলো দেখানো হয়েছে।

```
'get whatever in settings to the ListBox
  Private Sub Form1_Load(sender As Object, e As EventArgs) Handles Me.Load
    For Each item As String In My.Settings.PictureList
      ListBox1.Items.Add(item) 'my.settings items adding to listBox1
      Next
  End Sub
```

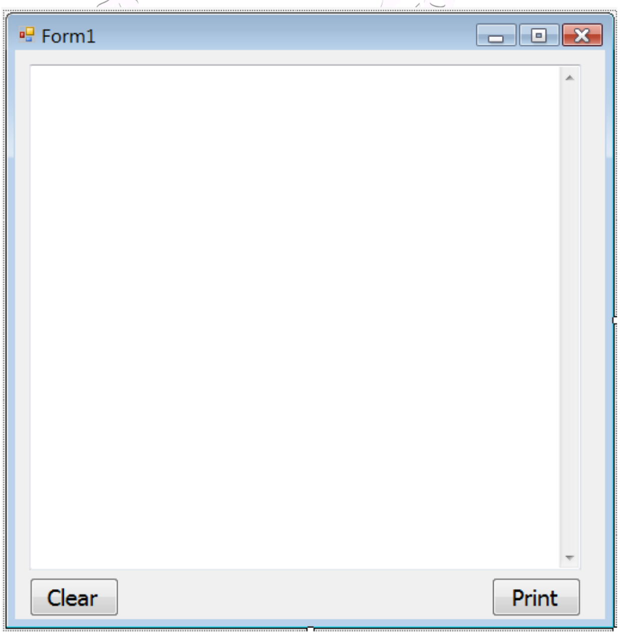
লক্ষ করুন, পিকচার লিস্টে যে ক'পি রেকর্ড আছে তা আমরা একে একে লিস্ট বাক্সে এনে প্রদর্শন করেছি। সুতরাং আলবামের সকল সংরক্ষিত ছবি পিকচার বাক্সে আনতে কোনো বাধা নেই। শুধুমাত্র লিংকে ক্লিক করলেই কাজ হবে। এবার দেখে নেবো ফর্ম ক্লোজিং সাবের কোড।

```
'save whatever in the list to setting as lines of strings
Private Sub Form1_FormClosing(sender As Object, e As FormClosingEventArgs) Handles Me.FormClosing
    My.Settings.PictureList = New Specialized.StringCollection() 'pictureList collection Instant declaration
        For Each item As String In ListBox1.Items
          My.Settings.PictureList.Add(item) 'ListBox items add to settings one by one
          My.Settings.Save() 'and save it
        Next
End Sub
```



এ কোডগুলোর প্রথম লাইন সেটিংয়ে সংরক্ষিত লিস্ট মুছে দেবে ও একটি নতুন কালেকশন তৈরি করবে। এরপর লিস্ট বাক্সের প্রতিটি লাইন একে একে ঐ কালেকশনে নিয়ে সংরক্ষণ করবে। সুতরাং আলবামের সকল লিংক সংরক্ষিত হবে। ব্যবহারকারী যখন প্রোগ্রাম আবার রান করবেন তখন ঐ কালেকশন থেকে লিংকগুলো উপরের ফর্ম লোড ইভেন্ট সাবের কোডের মাধ্যমে লিস্ট বাক্সে চলে আসবে। সুতরাং আমরা সত্যিকার অর্থে একটি ছবি আলবাম তৈরি করলাম। বায়ে এর রানটাইমকালীন একটি চিত্র দিয়েছি। দেখে নিন।

40. PrintDialog: এই ডায়লগ ও PrintDocument-অবজেক্ট একত্রে কাজ করে আপনার টেক্সট প্রিন্ট করতে পারেন ডিফোল্ট প্রিন্টারে। আমরা একটি উদাহরণ তুলে ধরছি।



একটি নতুন প্রজেক্ট তৈরি করুন। ফর্মের মধ্যে যুক্ত করুন দু'টি বোতাম ও একটি টেক্সট বাক্স। এরপর টুলবাক্স থেকে ডবল ক্লিক করে সলুশনে নিয়ে আসুন **PrintDialog** এবং PrintDocument অবজেক্ট। ডানের চিত্রটি দেখুন।
এবার ক্লিয়ার বোতামে ডবল ক্লিক করুন এবং সাবের ভেতর নিচের কোড লাইনটি লিখুন।

```
Private Sub Button2_Click(sender As Object, e As EventArgs)
Handles Button2.Click
    TextBox1.Text = ""
End Sub
```

এই কোড টেস্টট্ বাক্সকে খালি করে দেবে। এবার ডবল ক্লিক করুন

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

নিচের ধূসর এলাকায় পড়ে থাকা আমাদের PrintDocument1 -কে। একটি হ্যান্ডলার সাব আসবে ফর্মের কোড জানালায়। এতে নিচের কোডটুকু লিখুন।

```
'set the document font and colour
Private Sub PrintDocument1_PrintPage(sender As Object, e As Printing.PrintPageEventArgs) Handles
PrintDocument1.PrintPage
    Dim font1 As New Font("TonnyBanglaMJ", 14, FontStyle.Regular)
    e.Graphics.DrawString(TextBox1.Text, font1, Brushes.Black, 100, 100)
End Subr
```

এই কোডের কাজ হলো টেক্সট বাক্সের লেখাটুকুর ফন্ট, সাইজ, ফন্ট স্টাইল, ব্রাশ ইত্যাদি সেট করা। এবার প্রিন্ট বোতামে ডবল ক্লিক করুন। নিচের কোডটুকু লিখুন ইভেন্ট হ্যান্ডলারে।

```
'show setup dialog and then print using PrintDocument Object
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
   If TextBox1.Text = "" Then
      MsgBox("Please write something!")
      Exit Sub
    End If
    Dim result As DialogResult = PrintDialog1.ShowDialog()
    If result = 1 Then
      PrintDocument1.Print()
    End If
End Sub
```

এই কোডের কাজ হলো প্রথমে নিশ্চিত করা টেক্সট বাক্সে কিছু লেখা আছে। এরপর প্রিন্ট ডায়লগ দেখানো সেটিংয়ের জন্য। তারপর প্রিন্ট ডকুমেন্ট অবজেক্টকে এ সেটিং মুতাবিক প্রিন্ট করার জন্য বলা। ব্যস! এটুকুই। লক্ষ করুন, মাঝখানে ডিম স্টেটমেন্ট এর মাধ্যমে আমরা প্রিন্ট ডায়লগ প্রদর্শন করেছি। আমরা ব্যবহারকারীর একশন কি জানতে চাই। কারণ তিনি প্রিন্ট না করে ক্যানসেল বোতামেও ক্লিক করতে পারেন। প্রিন্ট বোতামের ফলাফল ১ এবং ক্যানসেল বোতামের ফলাফল ২। সুতরাং রেজাল্ট ১ হলেই প্রিন্টের নির্দেশ দিতে পারি। আপনার টেক্সট বাক্সে যাই থাকুক- এ ক্ষেত্রে বিজয় কিবোর্ডে বাংলায় লিখা থাকবে, তা-ই ডিফোল্ট প্রিন্টারে প্রিন্ট হবে। আমি টেস্ট করে দেখেছি। আপনিও দেখে নিন।

41. PrintDocument: আমরা এইমাত্র এ অবজেক্টের কার্যকারিতার ওপর আলোচনা করেছি উদাহরণসহ। সুতরাং এখানে আর কিছু বলার নেই।

42. PrintPreviewControl: প্রিন্টিংয়ের জন্য প্রিন্ট ডায়লগ ও প্রিন্ট ডকুমেন্টই যথেষ্ট। তবে সময় সময় আমরা পিন্ট্রের প্রিভিউ দেখতেও চাই। এই কম্পোনেন্টের কাজ এটাই।

43. PrintPreviewDialog: আমাদের সৃষ্ট উদাহরণটিকে কিছুটা পরিবর্তন করে নিলে এই ডায়লগের কার্যকারিতাও দেখতে পারি। এটার কাজও প্রিন্ট করা, তবে প্রথমে প্রদর্শিত হবে একটি প্রিভিউ। প্রিন্টিংসহ বেশ ক’টি ফাংশন আছে। আসুন, ইতোমধ্যে আলোচিত প্রজেক্টকে কিছুটা মডিফাই করি।

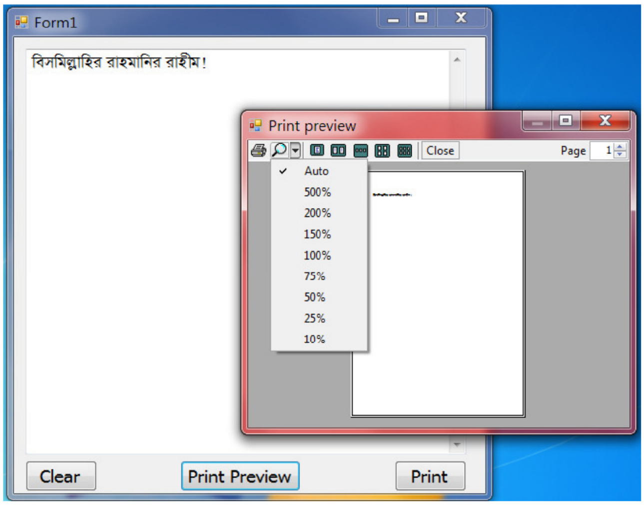
প্রথমেই আমরা টুলবাক্সে যেয়ে ফর্মে আরো একটি বোতাম ও **PrintPreviewDialog** যুক্ত করবো। বোতামের নামকরণ করবো: Print Preview। নতুন বোতামে ডবল ক্লিক করে সাবরুটিনে নিচের কোডটুকু লিখুন।

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    If TextBox1.Text = "" Then
      MsgBox("Write something!")
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
    Else
       PrintPreviewDialog1.Show()
    End If
End Sub
```

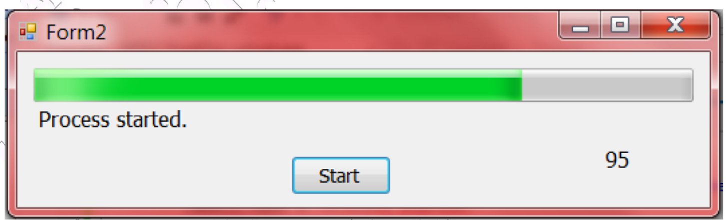


এই যা- আর কিছু লাগবে না। প্রিন্ট প্রিভিউ করেও এবার প্রিন্টিং করতে পারেন। ডানের চিত্রটি দেখুন।

44. Process: এটার কাজ হলো কম্পিউটারে চলন্ত প্রসেসগুলো সম্পর্কে তথ্য সংগ্রহ করা ও প্রয়োজনে প্রদর্শন করা। এটা মূলত সিস্টেম ডায়াগনস্টিক্স এর একটি ফাংশন। এডভান্স স্তরের প্রোগ্রামাররা একে ব্যবহার করতে পারেন। সুতরাং আমাদের ক্ষেত্রে এর ওপর আলোচনা আদৌ সঠিক হবে না।

45. ProgressBar: ওয়েভ ব্রাউজার উদহরণ প্রোগ্রাম তৈরির সময় আমরা এর ব্যবহার দেখেছি। কোনো দীর্ঘ প্রসেসের সময় ব্যবহারকারীকে ফিডব্যাক দিতে এটি কাজে লাগানো যায়। এর দু'টি গুরুত্বপূর্ণ প্রোপার্টি আছে: Minimum এবং Maximum। প্রথমাবস্থায় বারটি সম্পূর্ণ রংহীন থাকবে আর দ্বিতীয় অবস্থায় সম্পূর্ণরূপে রঙে পরিপূর্ণ হবে। একটি উদাহরণ দেখে নিই।

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Label1.Text = "Process started."
    For x As Integer = 0 To 100
       Label2.Text = x
       ProgressBar1.PerformStep()
       wait(1) 'wait for a millisecond
    Next
    ProgressBar1.Refresh()
    Label1.Text = "Process Ended."
End Sub
```




```
 ' Loops for a specificied period of time (milliseconds)
Private Sub wait(ByVal interval As Integer)
    Dim sw As New Stopwatch
    sw.Start()
    Do While sw.ElapsedMilliseconds < interval
       ' Allows UI to remain responsive
       Application.DoEvents()
    Loop
    sw.Stop()
End Sub
```

উপরের wait সাবরুটিন দ্বারা নির্দিষ্ট মিলিসেকেন্ড পরিমাণ অপেক্ষার ব্যবস্থা করেছি। এটা প্রগ্রেস বারের মাত্রা বৃদ্ধির পরই বার বার কল করেছি যাতেকরে বারের গতিটি দৃশ্যমান হয়।

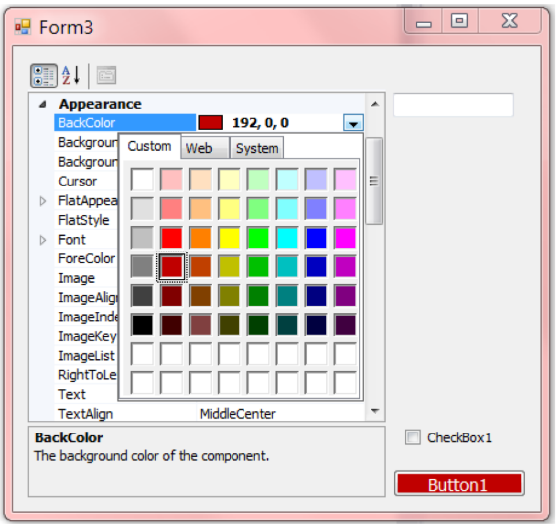
46. PropertyGrid: এটা আইডিই [Integrated Development Environment] এর মতো বস্তুর প্রোপার্টিগুলো দেখায়। কন্ট্রোলের দু'টি গুরুত্বপূর্ণ প্রোপার্টি হলো: ১. SelectedObject এবং ২. SelectedObjects। প্রথমটির অর্থ একটি অবজেক্ট ও দ্বিতীয়টির অর্থ একাধিক অবজেক্ট। রানটাইমে অবজেক্টের প্রোপার্টি রদবদলও করা যায়। আমরা একটি উদাহরণ দেখাবো।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

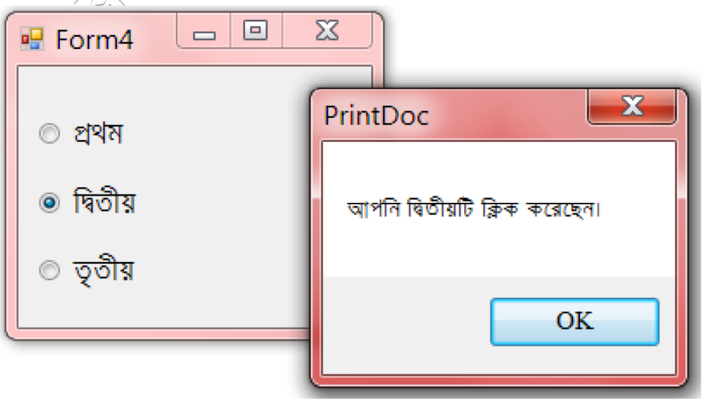
একটি উইন্ডোজ ফর্ম প্রজেক্ট তৈরি করে ফর্মের ভেতর একটি বোতাম, একটি চেক বাক্স, একটি টেক্সট বাক্স ও একটি প্রোপার্টি গ্রিড যুক্ত করুন। এরপর নিচের কোডগুলো বস্তুর ক্লিক ইভেন্ট সাবে লিখুন।

```
'make SelectedObject as Button1
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    PropertyGrid1.SelectedObject = Button1
End Sub
'make SelectedObject as CheckBox1
Private Sub CheckBox1_CheckedChanged(sender As Object, e As
EventArgs) Handles CheckBox1.CheckedChanged
    PropertyGrid1.SelectedObject = CheckBox1
End Sub
'make SelectedObject as TextBox1
Private Sub TextBox1_MouseEnter(sender As Object, e As EventArgs)
Handles TextBox1.MouseEnter
    PropertyGrid1.SelectedObject = TextBox1
End Sub
```



আমরা এসব কোড দ্বারা প্রোপার্টি গ্রিডের সিলেক্টেড অবজেক্ট প্রোপার্টি বস্তুর সাথে সম্পৃক্ত করেছি মাত্র। দেখুন ফলাফল।

47. RadioButton: এ কন্ট্রোলে থাকে একটি গোল বোমাত ও পাশে কিছু লেখা। ব্যবহারকারী গোল বোতামে ক্লিক করলে কেন্দ্রে রঙিন একটি বৃত্ত আসে। কয়েকটি বোতাম একই সঙ্গে রাখা যায়। কোনটি ক্লিক করা আছে তা কোডের মাধ্যমে জেনে প্রয়োজনীয় ক্রিয়া সম্পাদন করাই হলো এর মূল কার্যকারিতা। দেখুন নীচের চিত্রটি।



48. RichTextBox: আমরা ইতোমধ্যে এই কন্ট্রোল নিয়ে কাজ করেছি। এটা টেক্সট বাক্সের মতোই- তবে কিছু মৌলিক পার্থক্য আছে। যেমন, এই বাক্সে যে কোনো ফন্ট [ইংরেজি, বাংলাসহ] ব্যবহার করা যায়। টেক্সট ফরমাটিংও সম্ভব। দেখুন আমাদের সৃষ্ট 'বাংলা টেক্সট এডিটর' [পৃষ্ঠা ১৩৬]।

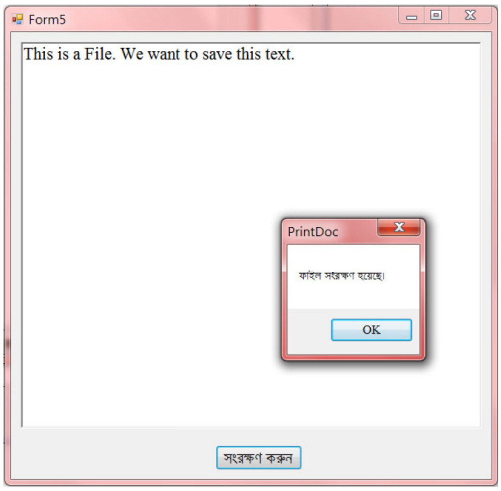
49. SaveFileDialog: এটা দ্বারা একটি সংরক্ষণ ডায়লগ দেখানো যায়। এটিও আমরা উক্ত 'বাংলা টেক্সট এডিটর প্রোগ্রামে ব্যবহার করেছি। তবে এখানে একটি ছোট্ট উদাহরণ দিচ্ছি। মনে করুন একটি জানালায় আছে রিচটেক্সবক্স কন্ট্রোল। আপনি এতে কিছু লিখে ফাইলে সংরক্ষণ করবেন। এটা আঞ্জাম দিতে প্রথমেই একটি ইউজার ইন্টারফেইস তৈরি করুন। দেখুন নিচের রানটাইমের চিত্রটি। এতে থাকবে একটি রিচটেক্সবক্স, একটি 'সংরক্ষণ করুন' বোতাম ও একটি সেইভফাইলডায়লগ। এবার দেখুন প্রোগ্রামের কোডগুলো।

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    If RichTextBox1.Text = "" Then
        Exit Sub
    End If
    SaveFileDialog1.DefaultExt = "*.rtf"
    SaveFileDialog1.Filter = "RTF Files|*.rtf"
    SaveFileDialog1.CreatePrompt = True
    If SaveFileDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
        RichTextBox1.SaveFile(SaveFileDialog1.FileName, RichTextBoxStreamType.RichText)
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
        MsgBox("ফাইল সংরক্ষণ হয়েছে।")
    End If
End Sub
```

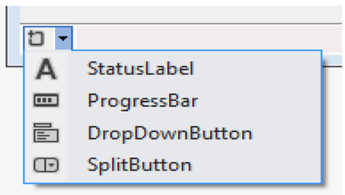


50. SerialPort: -এটা দ্বারা কম্পিউটারের যে কোনো ফিজিক্যাল সিরিয়েল পোর্টের বিভিন্ন প্রোপার্টি সম্পর্কে অবগত হওয়া যায়। এগুলোর মধ্যে আছে: baud rate, break signal, Data Set Ready (DSR) state, port name, parity, and stop bits ইত্যাদি। নোট করুন, এসব ব্যাপার অনেকটা এডভান্স স্টেজের প্রোগ্রামিংয়ের সঙ্গে জড়িত। সুতরাং এ গ্রন্থে এসব ব্যাপার নিয়ে আলোচনা আদৌ সঠিক হবে না।

51. ServiceController: এটাও উচ্চ পর্যায়ের প্রোগ্রামারদের সুবিধার্তে তৈরি করেছে মাইক্রসফট। উইন্ডোজ সার্ভিস প্রসেস ও মেথড এতে জড়িত। আমরা এ সম্পর্কে অতিরিক্ত কিছু বলতে পারবো না।

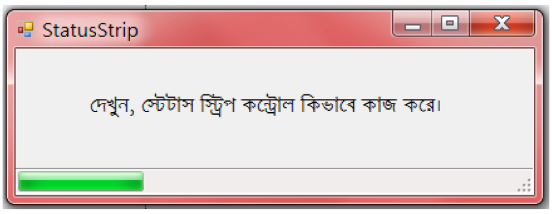
52. SplitContainer: এ কন্ট্রোল সম্পর্কে কিছুটা ব্যাখ্যা ১৭৭ পৃষ্ঠায় হয়েছে। এটি মূলত ডিজাইন কম্পোটেন্ট। এ সম্পর্কে এখানে আর কিছু বলছি না।

53. Splitter: এটার দ্বারাও ডিজাইনকালে কিছু কাজ করা যায়।



54. StatusStrip: এটা অনেকটা টুলস্ট্রিপের মতো। ফর্মের নিচে এটি প্রদর্শিত হয়। এর উদ্দেশ্য হলো স্টেটাস তথ্যাদি সংক্ষিপ্তভাবে দেখানো। এতে কয়েকটি বস্ত থাকে: স্টেটাস লেবেল, প্রগ্রেস বার, ড্রপডাউনবাটন এবং স্প্লিটবাটন। এগুলোর প্রতিটি আবার ফর্মের একেকটি কন্ট্রোল দ্বারা রেপ্রেজেন্ট করে। যেমন: প্রগ্রেসবারকে রেপ্রেজেন্ট করে টুলস্ট্রিপবার কন্ট্রোল। উদাহরণ থেকে ব্যাপারটি স্পষ্ট হবে। একটি ফর্মে যুক্ত করুন একটি স্টেটাস স্ট্রিপ কন্ট্রোল। ছোট্ট এ্যারোতে যেয়ে প্রথমে চয়েজ করুন: ProgressBar। ফর্মে ডবলক্লিক করে এর কোড এডিটর খুলুন। যুক্ত করুন নিচের কোডগুলো লোড ইভেন্ট সাবে:

```
While ToolStripProgressBar1.Value < 100
    ToolStripProgressBar1.Value += 5
End While
```



রান করলে বায়ের চিত্রটি দেখতে পাবেন।

55. TabControl: এই কন্ট্রোল পৃষ্ঠায় পৃষ্ঠায় গ্রুপকৃত তথ্যাদি প্রদর্শন করে থাকে। আমরা এর ব্যবহার ১৭৭ পৃষ্ঠায় দেখেছি। সুতরাং এখানে আর অতিরিক্ত কিছু বলার নেই- তবে! একটি উদাহরণ দেখে নিই।

একটি নতুন প্রজেক্ট তৈরি করুন। ফর্মের মধ্যে একটি ট্যাব কন্ট্রোল যুক্ত করুন। আমাদের উদাহরণে মোট ৫টি ট্যাব আছে। কন্ট্রোলের ডানের উপরের কোণে যে ছোট্ট এ্যারো আছে তাতে ক্লিক করে Abb Tab দ্বারা যতোটি প্রয়োজন ট্যাব যুক্ত করতে পারেন। প্রত্যেক ট্যাবের আলাদা জানালা আছে যাতে যুক্ত করা যায় অন্যান্য কন্ট্রোল। আমি প্রথম ট্যাবে একটি পিকচার বাক্স ও একটি লেবেল যুক্ত করেছি। পিকচার বাক্সে দেখাবো একটি ছবি এবং লেবেলে এ ছবি সম্পর্কে তথ্য থাকবে। অন্যান্য ট্যাবেও বিভিন্ন বস্ত যুক্ত করেছি। এবার দেখে নিন কোডটুকু।

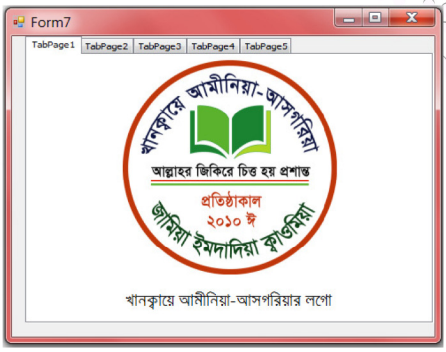
১৭তম পরিচ্ছেদ: *Windows Form Controls*

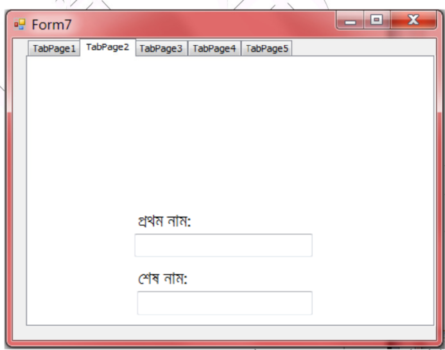
উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

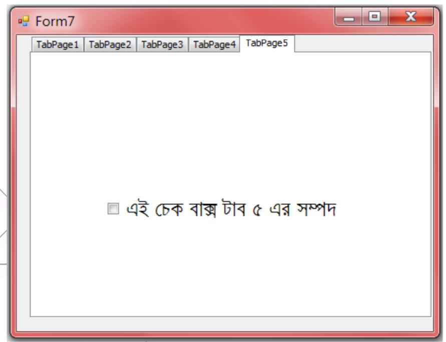
```
Private Sub TabControl_Load(sender As Object, e As EventArgs) Handles MyBase.Load
      ' select the tabpage #3
      TabControl1.SelectedIndex = 4
End Sub

Private Sub TabControl1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
TabControl1.SelectedIndexChanged
      If TabControl1.SelectedIndex = 0 Then
         PictureBox1.Image = My.Resources.LogoA
      End If
End Sub
```

উপরের প্রথম সাবরুটিনে আমরা ট্যাবের ডিফোল্ট ইনডেক্স কি হবে তা নির্দেশ করেছি। আর দ্বিতীয় সাব দ্বারা ইনডেক্স সিলেকশন ০ হলে ছবি প্রদর্শনের কথা বলেছি। ছবিটি অবশ্য My.Resources- এ ইতোমধ্যে এনে রেখেছিলাম। এবার দেখুন রানটাইমের ক'টি চিত্র।







56. TableLayoutPanel: এটা ডিজাইনকালে ব্যবহারের একটি কন্ট্রোল। এটার ওপর আলোচনা ইতোমধ্যে পৃষ্ঠা ১৭৭-এ হয়ে গেছে। সুতরাং এখানে আর কিছু বলার নেই।

57. TextBox: এ কন্ট্রোলটি সম্ভবত সর্বাধিক বেশি ব্যবহৃত হয় ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিংয়ে। পুরো এ বইটি জুড়ে টেক্সট বাক্স ব্যবহারের উদাহরণ আছে। এখানে অতিরিক্ত যা বলার তাহলো, এটা রিচটেক্সটবক্স থেকে অনেকটা সহজ, এর ভেতর যে কোনো একটিমাত্র ফন্ট ব্যবহার করা যায়, ফন্টের রঙ একই হতে হবে। বিশেষ তেমন কোনো ফর্মাটিংও সম্ভব নয়। মোটকথা বেশি সুবিধার জন্য রিচ টেক্সটবক্স কন্ট্রোল অনেকটা ব্যবহারোপযোগী। টেক্সবক্স কন্ট্রোলের গুরুত্বপূর্ণ ক'টি প্রোপার্টি হলো: MultiLine, MaxLength, ReadOnly, WordWrap, ScrollBars and Text ।

58. Timer: এটার কাজ হলো কোনো ফাংশন বা প্রসিডোরে ইন্টারভ্যল বা বিরতি দেওয়া। টাইমার কন্ট্রোল সলুশনের ধূসর এলাকায় পড়ে থাকে। একে কাজে লাগাতে হলে তার বিভিন্ন মেথড ও ফাংশন কল করতে হয় কোডে। একটি উদাহরণ দেখে নিই। নতুন একটি ফর্ম প্রজেক্ট তৈরি করুন। একটি টাইমার কন্ট্রোল প্রজেক্টে নিয়ে আসুন টুলবাক্স থেকে। আরো যুক্ত করুন দু'টি বোতাম ও একটি টেক্সট বাক্স। প্রথম বোতামের টেক্সট ভ্যালু হবে, 'টাইমার শুরু করুন' এবং দ্বিতীয়টির ভ্যালু হবে, 'টাইমার বন্ধ করুন'। টেক্সট বাক্সের ফন্ট সাইজ কিছুটা বাড়িয়ে দিন। এর ভ্যালু প্রপার্টি লিখুন ১। আমার মনে হয় উভয় বোতামের ফন্ট সাইজও কিছুটা বাড়িয়ে দিন যাতে স্পষ্ট দেখা যায়। টাইমারের প্রোপার্টিতে যেয়ে Interval প্রোপার্টির ভ্যালু লিখুন ১০০০। এর অর্থ ১০০০ মিলিসেকেন্ড বা ১ সেকেন্ড। এবার প্রথমে ডবল ক্লিক করুন টাইমার কন্ট্রোলে। কোড এডিটর জানালা আসবে। সেখানে Timer1_Tick ইভেন্ট সাবরুটিন। এ সাবের ভেতর লিখুন:

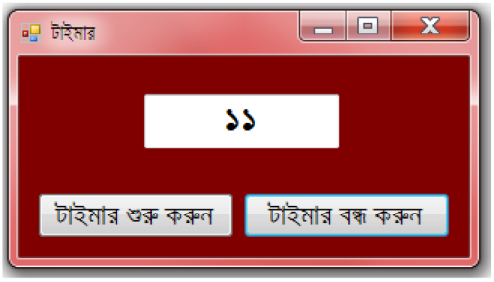
```
Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
    TextBox1.Text = Val(TextBox1.Text) + 1
End Sub
```

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

এরপর 'টাইমার শুরু করুন' বোতামে ডবল ক্লিক করে তার ইভেন্ট সাবে লিখুন:

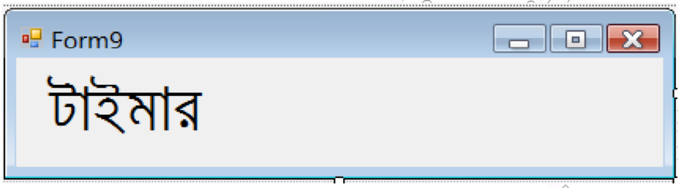
```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Timer1.Start()
End Sub
```



সবশেষে 'টাইমার বন্ধ করুন' বোতামে ডবল ক্লিক করে তার ইভেন্ট সাবে লিখুন:

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        Timer1.Stop()
End Sub
```

প্রোগ্রাম শুরু করে উভয় বোতামে ক্লিক করে দেখুন কিভাবে টেক্সট বাক্সে সংখ্যাটি বাড়ে ও বন্ধ হয়। আমরা এই টাইমার কন্ট্রোল দ্বারা একটি ডিজিটাল ঘড়ি বানাতে পারি। দেখুন এটা বানানো কতো সহজ!



নতুন প্রজেক্ট তৈরি করুন। ফর্মে একটি লেবেল নিয়ে আসুন। এর ফন্ট সাইজ ইত্যাদি পরিবর্তন করুন। যেভাবে বায়ের চিত্রে দেখানো হয়েছে। এর ফন্ট হবে বাংলা। আমরা বাংলা ডিজিটাল ঘড়ি বানাবো, কেমন?

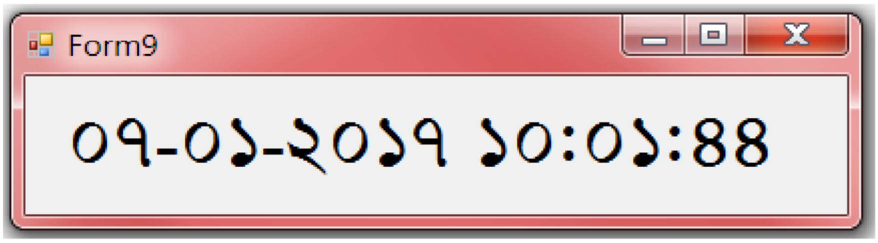
ফর্মের কোথাও ডবল ক্লিক করে এর লোড ইভেন্ড সাবরুটিন প্রদর্শন করুন। এতে লিখুন নিচের কোড লাইনটি। আমরা চাই প্রোগ্রাম শুরু হলেই টাইমার সক্রিয় হয়ে ওঠবে- তাই এই কোড।

```
Private Sub Form9_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Timer1.Enabled = True
End Sub
```

এবার টাইমারে ডবল ক্লিক করে তার ইভেন্ট সাব বের করুন। লিখুন এই কোডটুকু।

```
Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
    Label1.Text = Date.Now.ToString("dd-MM-yyyy hh:mm:ss")
    'Other date format "dd MM yyy hh mm ss"
End Sub
```

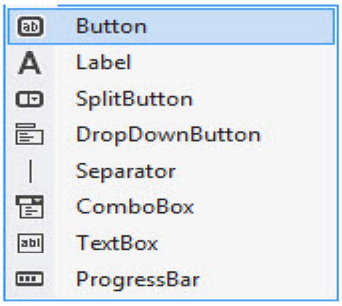
এ কোডের ওপর কিছুটা ব্যাখ্যার প্রয়োজন। আমরা লেবেলে তারিখ ও সময় দেখাতে চাই। সুতরাং এর টেক্সট-এ এখনকার তারিখ ও সময় [Now] প্রদর্শন করতে নির্দেশ দিচ্ছি। তারিখ ও সময় মূলত অবজেক্ট। ওগুলোকে স্ট্রিং বানাতে যেয়ে লিখতে হয়েছে ToSting কীওয়ার্ডটি। ব্রাকেটে আমরা তারিখ ও সময় ফরমাট কি হবে সেটা উল্লেখ করেছি। এবার রান করুন। নিচের চিত্রে আমরা ডিজিটাল ঘড়িটি দেখতে পাচ্ছি- আর এটি বাংলা সংখ্যা প্রদর্শন করছে!



১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

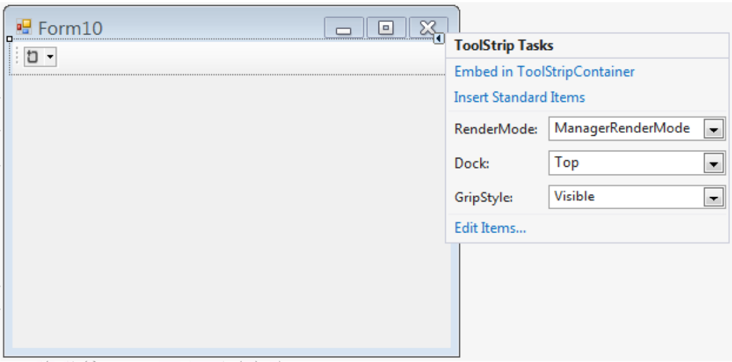
59. ToolStrip: এটার কাজ হলো একাধিক বোতাম, ড্রপডাউন এবং অন্যান্য টুল্স প্রদর্শন করা। এসব টুল্স সহজে ও তাড়াতাড়ি পেতে টুলস্ট্রিপ খুব উপযোগী। বার বার মেনু আইটেমে যাওয়ার প্রয়োজন নেই। নিচের লিস্টে দেখানো হয়েছে টুলস্ট্রিপে কোন্ কোন্ আইটেম থাকতে পারে।

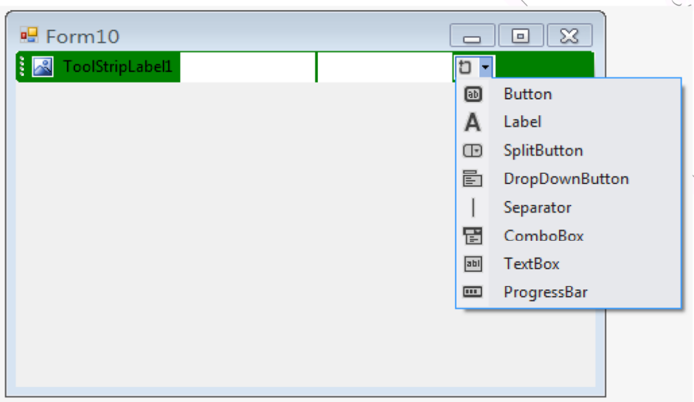


টুলস্ট্রিপ সৃষ্টির দু'টি সময় আছে: ১. ডিজাইনটাইম ও ২. রানটাইম। প্রথমে আমরা ডিজাইটাইমে টুলস্ট্রিপ তৈরির পদ্ধতি দেখবো।

একটি নতুন প্রজেক্ট তৈরি করুন। ফর্মের মধ্যে টুলবাক্স থেকে ডবল ক্লিক করে একটি টুলস্ট্রিপ যুক্ত করুন। এটা ফর্মের উপরে যেয়ে ডক করবে। ডানের ছোট্ট এ্যারোতে ক্লিক করে দেখতে পাবেন আরেকটি জানালা। নিচের চিত্রটি দেখুন।

এবার টুলস্ট্রিপে ক্লিক করে এর প্রোপার্টিতে যান। সেট BackColor - সবুজ এবং GripStyle - ভিজিবল। এরপর টুলস্ট্রিপের বায়ের ছোট্ট এ্যারোতে ক্লিক করুন। একটি লিস্ট আসবে [উপরের ছবি]। এ লিস্ট থেকে যে কোনো আইমে চয়েজ করতে পারেন। দেখুন নিচের চিত্রটি।

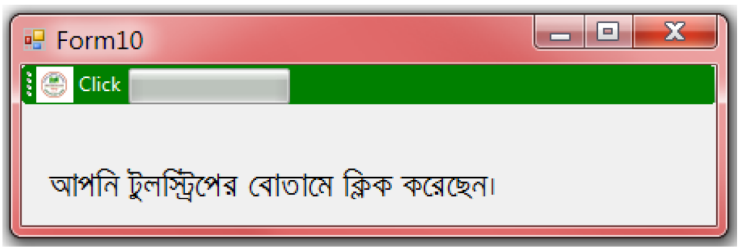




টুলস্ট্রিপ মূলত বিভিন্ন টুলস্ রাখার স্থান। আপনি এগুলোর ক্লিক ও অন্যান্য ইভেন্ট খুলে কোডিং করতে পারেন। যেমন নিচের কোডগুলো হলো যথাক্রমে বোতাম, লেবেল ও প্রগ্রেস বারের ক্লিক ইভেন্ট কোড। প্রজেক্ট রান করে টুলস্ট্রিপের যে কোন আইটেমে ক্লিক করুন।

```
Private Sub ToolStripButton1_Click(sender As Object, e As EventArgs) Handles ToolStripButton1.Click
    Label1.Text = "আপনি টুলস্ট্রিপের বোতামে ক্লিক করেছেন।"
End Sub
Private Sub ToolStripLabel1_Click_1(sender As Object, e As EventArgs) Handles ToolStripLabel1.Click
    Label1.Text = "আপনি টুলস্ট্রিপের লেবেলে ক্লিক করেছেন।"
End Sub
Private Sub ToolStripProgressBar1_Click(sender As Object, e As EventArgs) Handles ToolStripProgressBar1.Click
    Label1.Text = "আপনি টুলস্ট্রিপের প্রগ্রেস বারে ক্লিক করেছেন।"
End Sub
```

মোটকথা, আমরা চাইলে টুলস্ট্রিপে বিভিন্ন আইটেম রেখে এগুলোর বিভিন্ন ইভেন্টে কোড লিখে কাঙ্ক্ষিত ফলাফল পেতে পারি। এভাবে করলে ব্যবহারকারীর জন্য সুন্দর একটি ইন্টারফেইসও তৈরি হতে পারে।

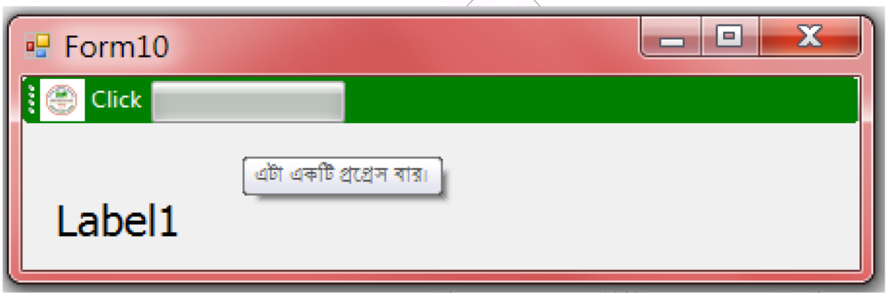


১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

60. ToolStripContainer: এটা একটি কন্টেইনার যার ভেতর ইতোমধ্যে আলোচিত টুলস্ট্রিপ থাকতে পারে। সুতরাং এটা ডিজাইনকালীন সুবিধার জন্য তৈরি হয়েছে।

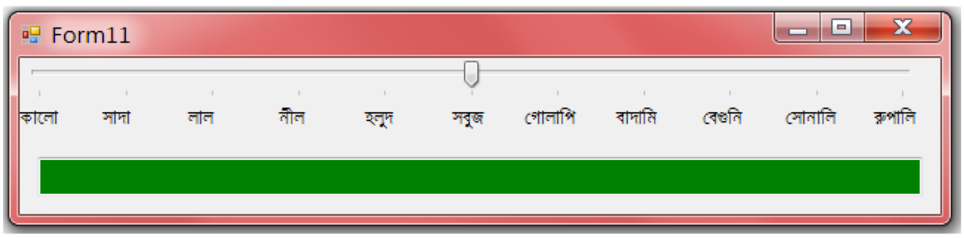
61. ToolTip: এ বস্তুটি খুব ব্যবারোপযোগী। অন্য বস্তু যেমন বোতাম, লেবেল ইত্যাদির উপর মাউস পয়েন্টার গেলেই কিছু লেখা আসবে, যাতে নির্দেশ থাকবে এ বস্তুটির উদ্দেশ্য কি। শিক্ষার্থীরা নিশ্চয়ই টুলটিপ দেখেছেন। দেখে নিই একটি উদাহরণ। আগের টুলস্ট্রিপ সম্পর্কিত প্রোগ্রামে আমরা টুলটিপ কন্ট্রোল যোগ করতে পারি। এবার দেখবেন ফর্মের সকল অন্যান্য কন্ট্রোলের প্রোপার্টিতে 'ToolTipText' নামক একটি অতিরিক্ত আইটেম এসেছে। আপনি এখানে যাই লিখবেন তা-ই আসবে যখনই মাউসটি ঐ কন্ট্রোলের ওপর যাবে। দেখুন নিচের চিত্রটি।



62. TrackBar: এটা অনেকটাই স্ক্রলবারের মতো। তবে এ ক্ষেত্রে ব্যবহারকারী ইচ্ছে মতো যে কোনো স্থানে স্ক্রল করতে পারেন ও সে মুতাবিক কোড করে বিভিন্ন ক্রিয়া সম্পাদনও সম্ভব। একটি সুন্দর উদাহরণ দেখে নিই। প্রথমে আমরা একটি ফর্ম ডিজাইন করবো। দেখুন নিচের রানকালীন চিত্রটি।

কোডিং: এটা খুব সহজ। প্রথমে ট্রাকবার১ এর ওপর ডবল ক্লিক করে কোড জানালা খুলুন। এরপর নিচের কোডগুলো লিখুন। আপনি সব টাইপ করতে হবে না। একটি করে বাকিগুলো কপি-পেস্ট করতে পারেন। পরিবর্তন হবে সংখ্যা ও রঙ।

```
Private Sub TrackBar1_Scroll(sender As Object, e As EventArgs) Handles TrackBar1.Scroll
    Dim tValue As Integer = TrackBar1.Value
    If tValue = 0 Then TextBox1.BackColor = Color.Black
    If tValue = 1 Then TextBox1.BackColor = Color.White
    If tValue = 2 Then TextBox1.BackColor = Color.Red
    If tValue = 3 Then TextBox1.BackColor = Color.Blue
    If tValue = 4 Then TextBox1.BackColor = Color.Yellow
    If tValue = 5 Then TextBox1.BackColor = Color.Green
    If tValue = 6 Then TextBox1.BackColor = Color.Pink
    If tValue = 7 Then TextBox1.BackColor = Color.Brown
    If tValue = 8 Then TextBox1.BackColor = Color.Purple
    If tValue = 9 Then TextBox1.BackColor = Color.Gold
    If tValue = 10 Then TextBox1.BackColor = Color.Silver
End Sub
```



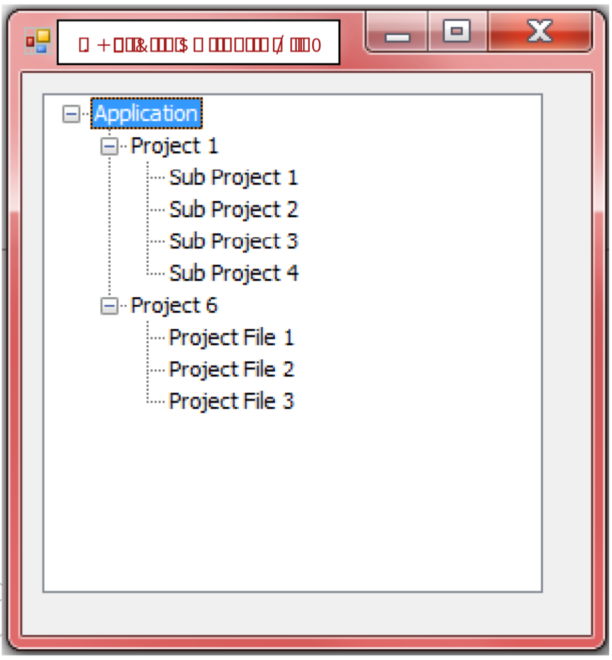
63. TreeView: এ কন্ট্রোল দ্বারা উইন্ডোজের এক্সপ্লোরারের মতো বিভিন্ন বস্তু দেখানো যায়। প্রতিটি নোডে থাকতে পারে আরো বাচ্চা নোড। একটি সাধারণ উদাহরণ থেকে এ সম্পর্কে আরো জানা যাবে। ট্রিভিউ কন্ট্রোল ডিজাইনটাইমে ফর্মে নেওয়া যায় বা প্রোগ্রাম ডোক দ্বারাও তৈরি করা যায়। আমরা দ্বিতীয় পদ্ধতিতে এটি তৈরি করে দেখাবো।

প্রথমেই একটি নতুন প্রজেক্ট তৈরি করুন। ফর্মের মধ্যে ডবল ক্লিক করে এর কোড এডিটর জানালা খুলুন। এরপর পরের পৃষ্ঠার কোডগুলো ফর্মের লোড ইভেন্ট সাবে লিখুন।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্স: *Windows Form Controls*

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    'create a new TreeView
    Dim TreeView1 As TreeView
    TreeView1 = New TreeView()
    TreeView1.Location = New Point(10, 10)
    TreeView1.Size = New Size(250, 250)
    Me.Controls.Add(TreeView1)
    TreeView1.Nodes.Clear()
    'Creating the root node
    Dim root = New TreeNode("Application")
    TreeView1.Nodes.Add(root)
    TreeView1.Nodes(0).Nodes.Add(New TreeNode("Project 1"))
    'Creating child nodes under the first child
    For loopindex As Integer = 1 To 4
      TreeView1.Nodes(0).Nodes(0).Nodes.Add(New _
        TreeNode("Sub Project" & Str(loopindex)))
    Next loopindex
    ' creating child nodes under the root
    TreeView1.Nodes(0).Nodes.Add(New TreeNode("Project 6"))
    'creating child nodes under the created child node
    For loopindex As Integer = 1 To 3
      TreeView1.Nodes(0).Nodes(1).Nodes.Add(New _
        TreeNode("Project File" & Str(loopindex)))
    Next loopindex
    ' Set the caption bar text of the form.
    Me.Text = "ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং"
  End Sub
```

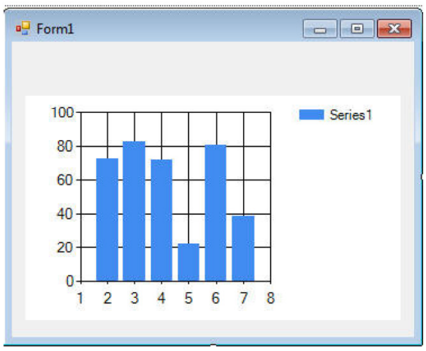


ডানের চিত্রে আমরা ট্রিভিউটি দেখতে পাচ্ছি।

64. VScrollBar: এটা হলো উপর-নিচ স্ক্রলবার কন্ট্রোল। এই কন্ট্রোল সম্পর্কে আমরা ইতোমধ্যে আলোচনা করেছি। দেখুন, পৃষ্ঠা ১৭৮।

65. WebBrowser: এই কন্ট্রোলটি হলো ওয়েভ ব্রাউজার তৈরির ক্লাস। আমরা একটি পূর্ণাঙ্গ ওয়েভ ব্রাউজার তৈরি করেছি। সুতরাং এ সম্পর্কে অতিরিক্ত কিছু বলার নেই। আপনি আবার দেখতে পারেন পৃষ্ঠা ১৬৩ থেকে ১৬৭।

66. Chart [Inside Data Sub Folder]: মনে করুন মাইক্রসফট এক্সেসের একটি ডাটাবেইজ আপনি তৈরি করলেন। সেখানে আছে দু'টি টেবিল: ১. Sales এবং ২. Expenses। আপনি চান, এই টেবিলদ্বয়ের একটি গ্রাফিক্যাল চার্ট আপনার বসকে দেখাবেন। এ কাজের জন্য ব্যবহার করতে পারেন ভিজুয়্যাল বেসিক.নেট এর Chart নামক কন্ট্রোলটি। আমরা এই উদাহরণ প্রোগ্রামটি করে নেবো।

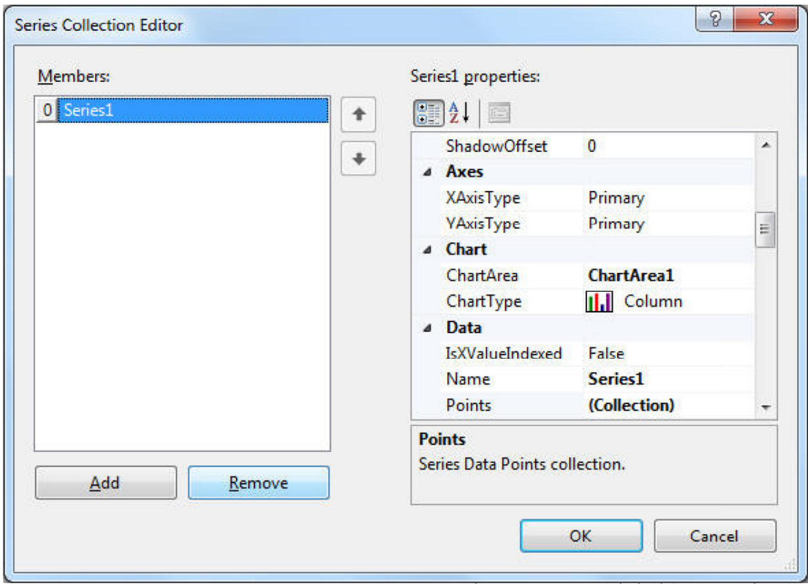


প্রথমে একটি প্রজেক্ট তৈরি করুন। এতে একটি মাত্র বোতাম টুলবাক্স থেকে নিয়ে আসুন। এর নামকরণ করতে পারেন: চার্ট তৈরি করুন। এবার টুলস্ বাক্স থেকে একটি চার্ট নিয়ে আসুন ফর্মে। টুলস্ বাক্সের Data নামক টাবটিতে ক্লিক করলেই Chart আইটেমটি দেখতে পাবেন। সাধারণত চার্টটি ডানের চিত্রের মতো প্রদর্শিত হবে ফর্মে। তবে আমরা এভাবে রাখবো না। আমরা সেল্স ও এক্সপেন্স কথাগুলো লিখবো। সুতরাং Series1 মুছে ফেলতে হবে।

১৭তম পরিচ্ছেদ: *Windows Form Controls*

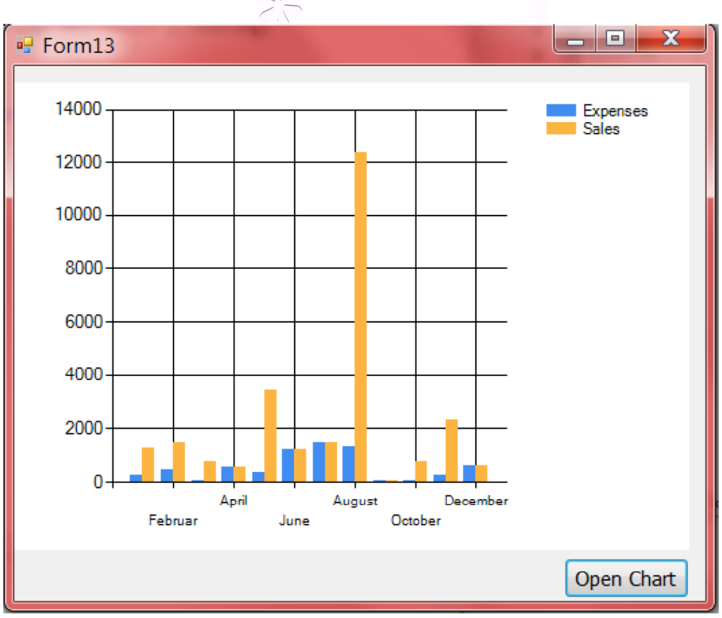
উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

প্রথমে চার্টে যেয়ে রাইট-ক্লিক করুন। চয়েজ করুন প্রোপার্টি। Series -এ যেয়ে ডানের [...] বোতামে ক্লিক করুন। যে জানালা আসবে তা নিচে দেখানো হয়েছে। Series সিলেক্ট করে Remove বোতামে ক্লিক করুন। ব্যস! অকে ক্লিক করে জানালা বন্ধ করুন। এবার 'চার্ট তৈরি করুন' বোতামে ডবল ক্লিক করে কোড এডিটর খুলুন। বোতামের ক্লিক ইভেন্টে নিচের কোডগুলো লিখুন।




```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Chart1.Series.Add("Expenses")
    Chart1.Series.Add("Sales")
    Dim Conn As OleDbConnection = New OleDbConnection
    Dim provider = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source ="
    Dim dataFile = "C:\Users\htci\Desktop\Sales.accdb" ' Change it to your Access Database location
    Conn.ConnectionString = provider & dataFile
    Conn.Open()
    Dim cmd As OleDbCommand = New OleDbCommand("SELECT [Month], [Amount] FROM [Expenses]", Conn)
    Dim dr As OleDbDataReader = cmd.ExecuteReader
    While dr.Read
        Chart1.Series("Expenses").Points.AddXY(dr("Month").ToString, dr("Amount").ToString)
    End While
    dr.Close()
    cmd.Dispose()
    cmd = New OleDbCommand("SELECT [Month], [Amount] FROM [Sales]", Conn)
    dr = cmd.ExecuteReader
    While dr.Read
        Chart1.Series("Sales").Points.AddXY(dr("Month").ToString, dr("Amount").ToString)
    End While
End Sub
```

প্রথমেই আমরা চার্টের সিরিজের মধ্যে দু'টি কথা লিখেছি। এগুলো চার্টের ডানে লিখা থাকবে। এর পরের লাইনে ডাটাবেইজের সঙ্গে যোগাযোগ সৃষ্টি করার জন্য নতুন কানেকশন ডিক্লায়ার করেছি। এরপর এক্সেসের ডাটাবেইজ খোলার যাবতীয় কোড এসেছে। যেখানে লেখা আছে: ' Change it to your Access Database location সেখানে আপনার এক্সেস ফাইলের প্যাথ স্ট্রিং লিখবেন। ব্যস! এই কোড রান করলেই আপনার ডাটাবেইজের চার্ট তৈরি হবে। দেখুন ডানের চিত্রটি।

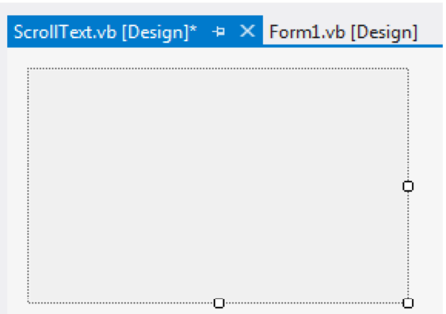


১৭তম পরিচ্ছেদ: *Windows Form Controls*

উইন্ডোজ ফর্ম কন্ট্রোল্‌স: *Windows Form Controls*

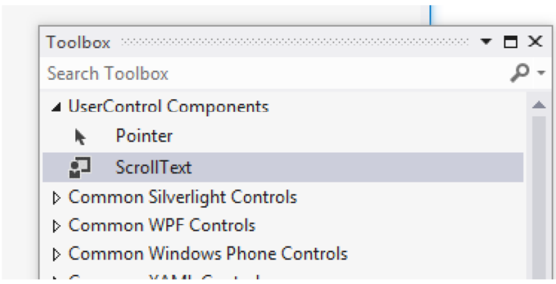
67. User Control: এতোক্ষণ আমরা ভিজুয়্যাল বেসিক.নেট এর তৈরি করা কন্ট্রোলগুলোর ওপর বিস্তারিত আলেচনা করেছি। তবে এখানেই শেষ নয়। ব্যাবহারকারী বা প্রোগ্রামাররা নিজেরাই নতুন কন্ট্রোল তৈরি করতে পারেন। এগুলোর মাধ্যমে হয়তো বিশেষ কোন কাজও হতে পারে। সুতরাং, আমরা এখন একটি ইউজার কন্ট্রোল তৈরির প্রক্রিয়ার ওপর আলেচনা করবো। বাস্তবে একটি নতুন কন্ট্রোল বানাবো।
মূলত তিন ধরনের ইউজার কন্ট্রোল আছে। যথা:
১. **Combination control:** এর অর্থ একাধিক ফর্ম কন্ট্রোল দ্বারা একটি নতুন কন্ট্রোল তৈরি করা। সকল উইন্ডোজ কন্ট্রোল এটির ভেতরে থাকবে এবং সবগুলো ফাংশন্যাল হবে।
২. **Inherited control:** এটা হলো এমন একটি কন্ট্রোল যা কোনো উইন্ডোজ কন্ট্রেল থেকে উত্তরাধিকার হিসাবে প্রাপ্ত। তবে এতে আরো থাকবে নতুন কিছু ফাংশন।
৩. **Custom control:** এটা সম্পূর্ণ নতুন কন্ট্রোল। এতে কোনো রেডিমেইড কন্ট্রোল থাকবে ন।

উপরে বর্ণিত তিনটির মধ্যে প্রথমটির ওপর একটি উদাহরণ এবার তুলে ধরছি। শিক্ষার্থীরা একে একে নিচের পদক্ষেপগুলো অনুসরণ করুন।
১. একটি ফর্ম প্রজেক্ট তৈরি করুন। এর নামকল যাচ্ছেতাই দিতে পারেন।
২. লোড হওয়ার পর চয়েজ করুন: Project > Add User Control। এর নামকরণ করুন: **ScrollText** ।
৩. এটি আসবে নিচের চিত্রের মতো। টুলবাক্স থেকে একটি লেবেল কন্ট্রোলে নিয়ে আসুন। এর নামকরণ করুন: LabelScroll। এরপর যুক্ত করুন একটি টাইমার। টাইমারের ওপর ডবল ক্লিক করে এর Tick ইভেন্ট সাব খুলুন। এই সাবের ভেতর লিখুন নিচের কোডগুলো।




```
Public Class ScrollText
   Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
      Dim strBegin As String = Mid(LabelScroll.Text, 1, 1) 'Extract First Character
      Dim strEnd As String = Mid(LabelScroll.Text, 2, LabelScroll.Text.Length()) 'Extract
Rest Of String
      LabelScroll.Text = strEnd & strBegin 'Add Beginning To End
   End Sub
End Class
```

লক্ষ করুন, এমনিতেই আমাদের কন্ট্রোলের নামে একটি ক্লাস এসেছে, ঠিক যেরূপ কোনো ফর্মের ক্লাস সাবরুটিন আসে। এই কোড একটি চলন্ত স্ট্রিং [লেখা] তৈরি করে। লেখার প্রথম অক্ষর ও বাকীগুলো যোগ করে এই কাজটি আঞ্জাম দেয়।
৪. এবার প্রোগ্রামটি বিল্ড করুন: Menu > Build > Build Solution। বিল্ড শেষে লক্ষ করুন, টুলবাক্সে আমাদের সৃষ্ট ScrollText কন্ট্রোলটিও User Control Components লিস্টে এসে গেছে। দেখুন নিচের চিত্রটি।



৫. এবার এই নতুন কন্ট্রোলটি ডবল ক্লিক করে আমাদের উইন্ডোজ ফর্মের ভেতর নিয়ে আসুন।
৬. ফর্মে দু'টি বোতাম যুক্ত করুন। এদের টেক্সট হবে যথাক্রমে: "শুরু" ও "বন্ধ"।
৭. 'শুরু' বোতামের ক্লিক সাবরুটিনে এই দু'টি লাইন লিখুন:

```
ScrollText1.LabelScroll.Text = "আমি তো ভাই খুব বেশি বেশি বেশি ক্ষুধার্ত..."
'Provide Text
ScrollText1.Timer1.Enabled = True 'Enable Timer
```

৭. 'বন্ধ' বোতামের ক্লিক সাবরুটিনে এই একটি লাইন লিখুন:

```
ScrollText1.Timer1.Enabled = False 'Stop Scrolling
```

ব্যস! আমাদের সৃষ্ট কন্ট্রোল এখন কাজ করবে। রান করে দেখুন।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

এটাই গ্রন্থের সর্বশেষ পরিচ্ছেদ। ভিজুয়্যাল বেসিক.নেট সম্পর্কে আমার যাকিছু শিখেছি তার ওপর সহজ একটি গাইড এ পরিচ্ছেদে তুলে ধরবো। শিক্ষার্থীরা নিজেদের এ্যাপলিকেশন সৃষ্টিতে এই পরিচেচ্ছদকে সময় সময় কাজে লাগাতে পারবেন। এছাড়া মাইক্রসফটের এই শকিক্তশালী প্রোগগ্রামিং ভাষার ওপরও এ গাইড থেকে অধিক বেশি জ্ঞানবান হবেন- এটাই আমার বিশ্বাস।

১. সার্বিক ধারণা: Overview

Microsoft Visual Basic.Net একটি অবজেক্ট অরিয়েন্টেড প্রোগগ্রামিং ভাষা। এর বাস্তবায়ন .নেট [বর্তমান ভার্শন: ৪.৫] নামক ফ্রেমওয়ার্কের ওপর নির্ভরশীল। ভাষার সবকিছুই মূলত বস্তুভিত্তিক। এর মৌলিক টাইপও [Short, Long, Integer, String, Boolean etc.] একেকটি বস্তু। .নেট দ্বারা সৃষ্ট প্রোগ্রাম উইনেন্ডাজ ছাড়াও লাইনাক্স [Linux] কিংবা ম্যাক অএসএক্স [Mac OSX] অপরেটিং সিস্টেমে রান করা যায়।

২. ডটনেট ফ্রেমওয়ার্ক: .Net Framework

এটা মাইক্রসফটের একটি বৈপ্লবিক সর্বাধুনিক প্রোগ্রামিং ফ্ল্যাটফর্ম। এটার দ্বারা প্রোগ্রামাররা সমগ্র বিশ্বব্যাপী প্রতিনিয়ত এ্যাপলিকেশন সৃষ্টি করে যাচ্ছেন:
(ক) উইন্ডোজ ডেস্কটপ এ্যাপলিকেশন।
(খ) ওয়েভ এ্যাপলিকেশন ও সার্ভিসেস।

ডটনেট এ্যাপলিকেশন বহুমূখি-ফ্লাটফর্মও বটে। এটি ডিজাইন করা হয়েছে বেশ ক'টি সফটওয়্যার ভাষার কথা চিন্তা করে। এগুলোর মধ্যে Visual Basic, C#, C++, Jscript, and COBOL ইত্যাদি এখানে উল্লেখযোগ্য। এ ভাষাগুলো ডটনেট ফ্রেমওয়ার্ক কাজে লাগাতে পারে এবং একে অন্যের সঙ্গে যোগাযোগও সৃষ্টি করতে পারে। ফ্রেমওয়ার্কের মধ্যে একটি বিরাট কোড লাইব্রেরি আছে যা উক্ত ভাষাগুলোর মাধ্যমে ব্যবহার করা যায়। এই গ্রন্থে আলোচিত Vb.Net মূলত লাইব্রেরির কোড ব্যবহারেরই প্রশিক্ষণ মাত্র।

৩. প্রোগ্রাম কাঠামো : Program Structure

যে কোন প্রোগ্রামে সাধারণত এই বিষয়গুলো থাকে: ক. নেইমস্পেইস ঘোষণা [Namespace declaration] উদাহরণ: Imports System, খ. ক্লাস বা মডিউল [Class or module] উদাহরণ: Public Class Form1; Module Module1, গ. প্রসিডিউর [Procedures- Sub, Function, Operator, Get, Set, AddHandler, RemoveHandler, RaiseEevent], ঘ. ভেরিয়েবল [Variables] উদাহরণ: Dim UserName as String, ঙ. মূল প্রসিডিউর [Main procedure] উদাহরণ: Sub Main(), চ. এক্সপ্রেশন ও স্টেটমেন্ট [Expressions & Statements] উদাহরণ: If MyVar = "Book" then; For x = 1 to 100 এবং ছ. মন্তব্য [Comments] উদাহরণ: 'This is the global variable for keeping User Name। শেষোক্ত [ছ] বিষয়টি বাস্তবে প্রোগ্রামের সঙ্গে সম্পৃক্ত নয়। এটা কম্পাইলার উপেক্ষা করে যাবে। তবে কমেন্ট থাকলে প্রোগ্রাম বুঝা অনেকটা সহজ হয়। ডিবাগিং ও নতুন সংস্করণ প্রকাশে এগুলো খুব সহায়ক হয়ে থাকে।

৪. ভিজুয়্যাল বেসিক.নেট মৌলিক সিনটেক্সট: VB.Net - Basic Syntax

আগেই বলেছি ভিজুয়্যাল বেসিক.নেট একটি পূর্ণাঙ্গ অবজেক্ট ওরিয়েন্ডেড ভাষা। এর অর্থ হলো কোনো প্রোগ্রামে থাকবে বিভিন্ন বস্তু বা অবজেক্ট। তারা একে অন্যের সঙ্গে মিথস্ক্রিয়া করবে কিছু ক্রিয়ার মাধ্যমে। আর এসব ক্রিয়ার নামই হচ্ছে পদ্ধতি বা মেথড। একই ধরনের অবজেক্টকে বলে একই টাইপ বা ক্লাসের। সুতরাং আমরা যখন কোনো ভিজুয়্যাল বেসিক.নেট প্রোগ্রাম নিয়ে আলোচনা করি তখন একে সংজ্ঞায়িত করতে পারি এভাবে: একগুচ্ছ অবজেক্ট বা বস্তু যারা একে অন্যের সঙ্গে যোগাযেগ সৃষ্টি করে যার তার মেথড বা পদ্ধতিসমূহের মাধ্যমে। এসব বস্তু ও মেথডের ওপর আমরা আরো

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

কিছু ব্যাখ্যা তুলে ধরতে পারি।
(ক) অবজেক্ট [Object]: অবজেক্টের আছে হাল বা স্টেট [State] ও চরিত্র [behavior]। উদাহরণ: একটি কুকুরের হালসমূহের মাধ্যে আছে: রং, নাম, ধরন। আরো আছে তার চরিত্র: লেজ নাড়ানো, প্রভুভক্তি, হাড্ডি খাওয়া, ডাকা, ছুটোছুটি করা, পান করা ইত্যাদি। অবজেক্ট হলো ক্লাসের ইন্সট্যান্ট [মুহূর্ত]।
(খ) ক্লাস [Class]: ক্লাসকে আমরা সংজ্ঞায়িত করতে পারি এভাবে: একটি ব্লু প্রিন্ট বা টেমপ্লেট যা ব্যাখ্যা করে এই টাইপের বস্তুর অবস্থা ও চরিত্র।
(গ) মেথড [Method]। এটা মূলত একটি চরিত্রকে বুঝায়। ক্লাসের মধ্যে অনেকগুলো মেথড থাকতে পারে। আর এসব মেথডের মধ্যেই আমরা লজিক বা যুক্তিগুলো লিখে থাকি। এগুলোর মধ্যেই তথ্যাদি পরিবর্তন, পরিবর্ধন, কাজে লাগানো ও সকল ক্রিয়া সম্পাদিত হয়।
(ঘ) মুহূর্ত ভেরিয়েবল [Instant variables]: প্রতিটি অবজেক্টেরই আছে তার নিজস্ব এক বা ততোধিক ভেরিয়েবল। এই মুহূর্ত ভেরিয়েবলসমূহে [অর্থাৎ অস্থায়ী কন্টেইনারে] ভ্যালু বা মান নির্দিষ্ট [assignment] করেই অবজেক্টের State সৃষ্টি করা হয়।

৫. পরিচায়ক: Identifiers

প্রোগ্রামার দ্বারা ডিফাইন করা যে কোনো ক্লাস, ভেরিয়েবল, ফাংশন, সাবরুটিন কিংবা যে কোনো বস্তুর পরিচায়ক হিসাবে বিশেষ নাম থাকতে হয়। এ নামকেই বলে আইডেনিন্টফাইয়ার। নামগুলো হতে হবে একক। এছাড়া ভিজুয়্যাল বেসিক.নেট ভাষার কিছু নাম আছে যেগগুলো রিজর্ভ। এগুলো প্রোগ্রামে আইডেন্টিফাইয়ার হিসাবে ব্যবহার করা নিষিদ্ধ। আমরা একটু পরই রিজার্ভ কিওয়ার্ডগগুলোর একটি লিস্ট তুলে ধরছি। আইডেন্টিফাইয়ার কিভাবে লিখবেন তারও কিছু নিয়মকানুন আছে। আমরা সেগুলো প্রথমে উল্লেখ করছি।
ক. নামের প্রথম অক্ষর কোনো সংখ্যা দেওয়া যাবে না। তবে আন্ডারস্কোর [_] গ্রহণযোগ্য। নামের মাঝখানে শূন্যস্থান [স্পেইস] দেওয়া যাবে না। যেমন: Dim _MyVariable; Dim FirstNumber1 - উভয়টি গ্রহণযোগ্য। কিন্তু, Number One কিংবা Dim 3TheNumber- গ্রহণযোগ্য নয়।
খ. আইডেন্টিফাইয়ারে এই ক্যারাক্টারগুলো দেওয়া যাবে না: ? - +! @ # % ^ & * ( ) [ ] { } . ; : " ' / \| ইত্যাদি।

নিচের টেবিলে আমরা রিজার্ভ কিওয়ার্ডগুলো তুলে ধরেছি।

AddHandler	AddressOf	Alias	And	AndAlso	As	Boolean
ByRef	Byte	ByVal	Call	Case	Catch	CBool
CByte	CChar	CDate	CDec	CDbl	Char	CInt
Class	CLng	CObj	Const	Continue	CSByte	CShort
Decimal	Declare	Default	Delegate	Dim	DirectCast	Do
Double	Each	Else	ElseIf	End	End If	Enum
Erase	Error	Event	Exit	False	Finally	For
Friend	Function	Get	GetType	GetXML Namespace	Global	GoTo
Handles	If	Implements	Imports	In	Inherits	Integer
Interface	Is	IsNot	Let	Lib	Like	Long
Loop	Me	Mod	Module	MustInherit	MustOverride	MyBase
MyClass	Namespace	Narrowing	New	Next	Not	Nothing
Not Inheritable	Not Overridable	Object	Of	On	Operator	Option
Optional	Or	OrElse	Overloads	Overridable	Overrides	ParamArray

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

Partial	Private	Property	Protected	Public	RaiseEvent	ReadOnly
ReDim	REM	Remove Handler	Resume	Return	SByte	Select
Set	Shadows	Shared	Short	Single	Static	Step
Stop	String	Structure	Sub	SyncLock	Then	Throw
To	True	Try	TryCast	TypeOf	UInteger	While
Widening	With	WithEvents	WriteOnly	Xor		

উপরোক্ত টেবিলের সকল শব্দ মূলত ভিজুয়্যাল বেসিক.নেট ভাষার সঙ্গে সম্পৃক্ত। সুতরাং এগুলো আইডেন্টিফাইয়ার হিসাবে ব্যবহার করা যাবে না।

৬. তথ্যাদির ধরন: Data Types

সকল প্রোগ্রামিং ভাষার মতো ভিজুয়্যাল বেসিক.নেটে আমরা সব ধরনের ডাটা টাইপ ব্যবহার করতে পারি। গ্রন্থের শুরুর দিকে আমরা এসব টাইপ নিয়ে আলোচনা করেছি। এখানে শুধুমাত্র লিস্টটি পূর্ণাঙ্গভাবে তুলে ধরার প্রয়োজন মনে করছি।

Data Type	Storage Allocation	Value Range
Boolean	Depends on implementing platform	**True** or **False**
Byte	1 byte	0 through 255 (unsigned)
Char	2 bytes	0 through 65535 (unsigned)
Date	8 bytes	0:00:00 (midnight) on January 1, 0001 through 11:59:59 PM on December 31, 9999
Decimal	16 bytes	0 through +/- 79,228,162,514,264,337,593,543,950,335 (+/- 7.9...E+28) with no decimal point; 0 through +/ -7.9228162514264337593543950335 with 28 places to the right of the decimal
Double	8 bytes	-1.79769313486231570E+308 through -4.94065645841246544E-324, for negative values 4.94065645841246544E-324 through 1.79769313486231570E+308, for positive values
Integer	4 bytes	-2,147,483,648 through 2,147,483,647 (signed)
Long	8 bytes	-9,223,372,036,854,775,808 through 9,223,372,036,854,775,807(signed)
Object	4 bytes on 32-bit platform 8 bytes on 64-bit platform	Any type can be stored in a variable of type Object
SByte	1 byte	-128 through 127 (signed)
Short	2 bytes	-32,768 through 32,767 (signed)
Single	4 bytes	-3.4028235E+38 through -1.401298E-45 for negative values; 1.401298E-45 through 3.4028235E+38 for positive values

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

Data Type	Storage Allocation	Value Range
String	Depends on implementing platform	0 to approximately 2 billion Unicode characters
UInteger	4 bytes	0 through 4,294,967,295 (unsigned)
ULong	8 bytes	0 through 18,446,744,073,709,551,615 (unsigned)
User-Defined	Depends on implementing platform	Each member of the structure has a range determined by its data type and independent of the ranges of the other members
UShort	2 bytes	0 through 65,535 (unsigned)

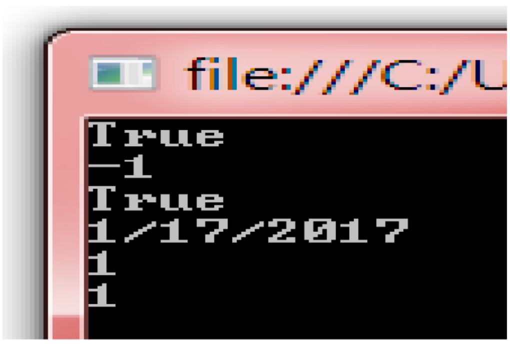
৭. টাইপ রূপান্তর ফাংশন: Types Conversion Functions

ভিজুয়্যাল বেসিক.নেট নিচের টেবিলে উল্লেখিত ফাংশনগুলো প্রোগ্রামারদের সুবিধার্থে তৈরি করে দিয়েছে।

S.N	Functions & Description	S.N	Functions & Description
1	**CBool(expression)** Converts the expression to Boolean data type.	9	**CObj(expression)** Converts the expression to Object type.
2	**CByte(expression)** Converts the expression to Byte data type.	10	**CSByte(expression)** Converts the expression to SByte data type.
3	**CChar(expression)** Converts the expression to Char data type.	11	**CShort(expression)** Converts the expression to Short data type.
4	**CDate(expression)** Converts the expression to Date data type	12	**CSng(expression)** Converts the expression to Single data type.
5	**CDbl(expression)** Converts the expression to Double data type.	13	**CStr(expression)** Converts the expression to String data type.
6	**CDec(expression)** Converts the expression to Decimal data type.	14	**CUInt(expression)** Converts the expression to UInt data type.
7	**CInt(expression)** Converts the expression to Integer data type.	15	**CULng(expression)** Converts the expression to ULng data type.
8	**CLng(expression)** Converts the expression to Long data type.	16	**CUShort(expression)** Converts the expression to UShort data type.

উদাহরণ: নিচের কোড ব্লকটি একটি কনসোল প্রজেক্ট তৈরি করে মডিউলে লিখে রান করুন।

```
Module Module1
    Sub Main()
        Dim n As Integer
        Dim da As Date
        Dim bl As Boolean = True
        n = 1234567
        da = Today
        Console.WriteLine(bl)
        Console.WriteLine(CSByte(bl))
        Console.WriteLine(CStr(bl))
        Console.WriteLine(CStr(da))
        Console.WriteLine(CChar(CChar(CStr(n))))
        Console.WriteLine(CChar(CStr(da)))
        Console.ReadKey()
    End Sub
End Module
```



বায়ের কনসোল প্রোগ্রাম রানের ফলাফল।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

৮. ভিজুয়্যাল বেসিক.নেট ভেরিয়েবল : Visual Basic.Net Variables

ভেরিয়েবল কম্পিউটারের স্টোরেজ মেমোরিতে একটি নাম ছাড়া কিছুই নয়। আমরা এই নামে বিভিন্ন তথ্য সংরক্ষণ করি ও প্রয়োজনে বের করে পরিবর্তন করতে পারি। প্রত্যেক ভেরিয়েবল অবশ্য কোনো বিশেষ ধরনের হতে হবে। একে বলে ভেরিয়েবলের টাইপ। কোন্ টাইপের ভেরিয়েবল তা-ই নির্ধারণ করে কী পরিমাণ মেমোরির প্রয়োজন। ইতোমধ্যে আমার ভেরিয়েবল টাইপ নিয়ে আলোচনা করেছি। মৌলিক টাইপগুলোকে আমরা নিচের টেবিলে দেওয়া পদ্ধতিতে শ্রেণিভুক্ত করতে পারি।

Type	Example
Integral types	SByte, Byte, Short, UShort, Integer, UInteger, Long, ULong and Char
Floating point types	Single and Double
Decimal types	Decimal
Boolean types	True or False values, as assigned
Date types	Date

৯. ভেরিয়েবল ডিক্লারেশন ও ইনিশিয়েলাইজেশন: Variable Declaration and Initialization

শিক্ষার্থীরা ভেরিয়েবল ডিক্লারেশন ও ইনিশিয়েলাইজেশন সম্পর্কে ওয়াকিফহাল আছেন। এরপরও নিচে পুর্ণাঙ্গ নিয়ম-কানুন তুলে ধরছি রেফারেন্স হিসাবে।

Syntax: [< attribute list>] [access modifier] [[Shared] [Shadows] | [Static]] [ReadOnly] Dim [WithEvents] variable list

উপরের সিনট্যাক্সের প্রতিটি শব্দের ব্যাখ্যা:

attribute list: এটা হলো ভেরিয়েবলের সঙ্গে সম্পর্কিত গুণাবলীর তালিকা- এটা অপশন্যাল।

access modifier: ভেরিয়েবলের প্রাপ্তির ব্যাপ্তি কতটুকু হবে। যেমন: Public, Protected, Friend, Protected Friend and Private। এটাও অপশন্যাল।

Shared: এটা ডিক্লায়ার করে যে ভেরিয়েবলটি সকল স্ট্রাকচার ও ক্লাসে ব্যবহারযোগ্য। এটা অপশন্যাল।

Shadows: এটা নির্দেশ করে যে ভেরিয়েবলটি পুনরায় একই নামে বেইজ ক্লাসে ডিক্লায়ার করা আছে। এটা অপশন্যাল।

Static: যে প্রসিডিউরে এটি ডিক্লায়ার হয়েছে সেটি বন্ধ হওয়ার পরও ভেরিয়েবলটি অপরিবর্তিত থাকবে। এটা অপশন্যাল।

ReadOnly: এর অর্থ ভেরিয়েবলটি শুধুমাত্র পড়া যাবে- কিন্তু লেখা যাবে না। এটা অপশন্যাল।

Dim: এর মানে ডাইমেনশন। এটা বাধ্যতামূলক।

WithEvents: ভেরিয়েবলে যে ইন্সস্ট্যান্ট [মুহূর্ত] এসাইন করা হয়েছে তা এর দ্বারা সক্রিয় হতে পারে। অপশন্যাল।

variable list: এতে থাকে ডিক্লায়ার করা ভেরিয়েবল তালিকা। বাধ্যতামূলক।

প্রত্যেক ভেরিয়েবল লিস্টে থাকবে:

variable name[([bounds list])] [As [New] data type] [= initializer]

variable name: এটা হচ্ছে ভেরিয়বলের নাম। বাধ্যতামূলক।

bounds list: এ্যারে ভেরিয়েবলের প্রত্যেক বিস্তৃতির ব্যাপ্তি ডিক্লায়ার করে। এটা অপশন্যাল।

As: কিওয়ার্ড। এটা থাকতে হবে।

New: এটা ক্লাসের নতুন একটি ইন্সস্ট্যান্ট তৈরি করে যখন ডাইমেনশন [Dim] স্টেটম্যান্ট রান করে। ঐচ্ছিক।

data type: অপশন স্ট্রিক্ট [Option Strict] অন থাকলে এটা বাধ্যতামূলক।

initializer: যদি নিউ [New] লেখা না থাকে, এটা ব্যবহার অপশন্যাল।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

কিছু গ্রহণযোগ্য ডিক্লারেশের উদাহরণ:

```
Dim StudentID As Integer
Dim StudentName As String
Dim Salary As Double
Dim count1, count2 As Integer
Dim status As Boolean
Dim exitButton As New System.Windows.Forms.Button
Dim lastTime, nextTime As Date
```

ভেরিয়েবল ইনিশিয়েলাইজেশন হয় নামের পরে সমান চিহ্ন [=] ও মান লিখে। যেমন: variable_name = value;

উদাহরণ:

```
Dim pi As Double
pi = 3.14159
```

প্রয়োজনে ডিক্লারেশনকালেও আমরা ভেরিয়েবল ইনিশিয়েলাইজ করতে পারি। যেমন:

```
Dim StudentID As Integer = 100
Dim StudentName As String = "Asif Talukder"
```

অপরিবর্তনীয় ভেরিয়েবল ডিক্লারেশন: Declaring Constants : ভিজুয়্যাল বেসিক.নেটে যে কোনো স্থানে ভেরিয়েবলের মতো কন্সস্ট্যান্ট ভেরিয়েবল ডিক্লায়ার করা যায়। এগুলো ঠিক ভেরিয়েবলের মতোই, কিন্তু এদের মান পরিবর্তন হয় না। একটি উদাহরণ দেখি।

```
Module constantsNenum
  Sub Main()
    Const PI = 3.14149 'here is the constant declaration
    Dim radius, area As Single
    radius = 7
    area = PI * radius * radius
    Console.WriteLine("Area = " & Str(area))
    Console.ReadKey()
  End Sub
End Module
```

উপরের প্রোগ্রাম রানের ফলাফল: Area = 153.933।

১০. প্রিন্ট ও প্রদর্শন কন্সস্ট্যান্ট: Print and Display Constants

ভিজুয়্যাল বেসিক.নেট আমাদেরকে নিচের কন্সস্ট্যান্টগুলো তৈরি করে দিয়েছে।

Constant	Description
vbCrLf	Carriage return/linefeed character combination.
vbCr	Carriage return character.
vbLf	Linefeed character.
vbNewLine	Newline character.
vbNullChar	Null character.
vbNullString	Not the same as a zero-length string (""); used for calling external procedures.
vbObjectError	Error number. User-defined error numbers should be greater than this value. For example: Err.Raise(Number) = vbObjectError + 1000
vbTab	Tab character.
vbBack	Backspace character.

১৮তম পরিচ্ছেদ: ***Visual Basic.Net: Simple Guide***

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

১১. লিস্ট ডিক্লারেশন: Declaring Enumerations

একসাথে একাধিক ভেরিয়েবল ডিক্লারেশন করার একটি পদ্ধতির নাম ইনুমারেশন **[Enumeration]**। আমরা চাইলে Enum .... End Enum ব্লকের ভেতর ভেরিয়েবল লিস্ট ডিক্লায়ার করতে পারি। যেমন:

```
Module constantsNenum
   Enum Colors
      red = 1
      orange = 2
      yellow = 3
      green = 4
      azure = 5
      blue = 6
      violet = 7
   End Enum
   Sub Main()
      Console.WriteLine("The Color Red is : " & Colors.red)
      Console.WriteLine("The Color Yellow is : " & Colors.yellow)
      Console.WriteLine("The Color Blue is : " & Colors.blue)
      Console.WriteLine("The Color Green is : " & Colors.green)
      Console.ReadKey()
   End Sub
End Module
```

বায়ের কনসোল প্রোগ্রামটি রান করলে আমরা যে ফলাফল পাবো তাহলো:

```
The Color Red is: 1
The Color Yellow is: 3
The Color Blue is: 6
The Color Green is: 4
```

১২. স্টেটমেন্ট: Statements

ইতোমধ্যে এ শব্দটি একাধিকবার উল্লেখিত হয়েছে। ভিজিুয়্যাল বেসিক.নেটে শব্দটির গুরুত্ব বিরাট। কোডের মধ্যে পূর্ণাঙ্গ নির্দেশনার নাম স্টেটমেন্ট। এতে থাকতে পারে: keywords, operators, variables, literal values, constants and expressions। মূলত দু'ধরনের স্টেটমেন্ট আছে: (১) ডিক্লারেশন স্টেটমেন্ট: এগুলো হলো যাদের মধ্যে থাকবে ভেরিয়েবলের নাম, কন্সস্ট্যান্ট বা প্রসিডোর- এতে ডাটা টাইপ ও উল্লেখিত হতে পারে। (২) এক্সেকিউটেবল স্টেটমেন্ট: এগুলো ঐরূপ যারা কোনো ক্রিয়া শুরু করে। এগুলো কোনো মেথড বা ফাংশন কল করতে পারে, কোড ব্লকে লুপিং ও ব্রাঞ্চিং করতে পারে। এ ধরনের স্টেটমেন্ট দ্বারা মান কিংবা এক্সপ্রেশন ভেরিয়েবল বা কন্সস্ট্যান্টে এসাইন করা যায়। আর এরূপ করাকে বলে এসাইনমেন্ট স্টেটমেন্ট। আমরা এই উভয় প্রকার স্টেটমেন্ট এর ওপর এখন আলাদাভাবে ব্যাখ্যা তুলে ধরছি।

(১) ডিক্লারেশন স্টেটমেন্ট: Declaration Statements

এই স্টেটমেন্ট ব্যবহৃত হয় প্রসিডোর, ভেরিয়েবল, প্রোপার্টি, এ্যারে ও কন্সস্ট্যান্ট এর নাম ও ডিক্লারেশনের জন্য। নিচের টেবিলে ভিজুয়্যাল বেসিক.নেট এর ডিক্লারেশন স্টেটমেন্টগুলো তুলে ধরেছি উদাহরণসহ।

S.N	Statements and Description	Example
1	**Dim Statement** Declares and allocates storage space for one or more variables.	Dim number As Integer Dim quantity As Integer = 100 Dim message As String = "Hello!"
2	**Const Statement** Declares and defines one or more constants.	Const maximum As Long = 1000 Const naturalLogBase As Object = CDec(2.7182818284)

পরের পৃষ্ঠায় বাকি টেবিল ... ...

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

S.N	Statements and Description	Example
3	**Enum Statement** Declares an enumeration and defines the values of its members.	Enum CoffeeMugSize Jumbo ExtraLarge Large Medium Small End Enum
4	**Class Statement** Declares the name of a class and introduces the definition of the variables, properties, events, and procedures that the class comprises.	Class Box Public length As Double Public breadth As Double Public height As Double End Class
5	**Structure Statement** Declares the name of a structure and introduces the definition of the variables, properties, events, and procedures that the structure comprises.	Structure Box Public length As Double Public breadth As Double Public height As Double End Structure
6	**Module Statement** Declares the name of a module and introduces the definition of the variables, properties, events, and procedures that the module comprises.	Public Module myModule Sub Main() Dim user As String = InputBox("What is your name?") MsgBox("User name is" & user) End Sub End Module
7	**Interface Statement** Declares the name of an interface and introduces the definitions of the members that the interface comprises.	Public Interface MyInterface Sub doSomething() End Interface
8	**Function Statement** Declares the name, parameters, and code that define a Function procedure.	Function myFunction (ByVal n As Integer) As Double Return 5.87 * n End Function
9	**Sub Statement** Declares the name, parameters, and code that define a Sub procedure.	Sub mySub(ByVal s As String) Return End Sub
10	**Declare Statement** Declares a reference to a procedure implemented in an external file.	Declare Function getUserName Lib "advapi32.dll" Alias "GetUserNameA" (ByVal lpBuffer As String, ByRef nSize As Integer) As Integer
11	**Operator Statement** Declares the operator symbol, operands, and code that define an operator procedure on a class or structure.	Public Shared Operator + (ByVal x As obj, ByVal y As obj) As obj Dim r As New obj ' implemention code for r = x + y Return r End Operator

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

S.N	Statements and Description	Example
12	**Property Statement** Declares the name of a property, and the property procedures used to store and retrieve the value of the property.	ReadOnly Property quote() As String Get Return quoteString End Get End Property
13	**Event Statement** Declares a user-defined event.	Public Event Finished()
14	**Delegate Statement** Used to declare a delegate.	Delegate Function MathOperator(ByVal x As Double, ByVal y As Double) As Double

(২) এক্সেকিউটেবল স্টেটমেন্ট: Executable Statements

এরূপ স্টেটমেন্টের কাজ হলো ক্রিয়া সম্পাদন করা। যেসব স্টেটমেন্ট অন্য প্রসিডোর বা ফাংশন কল করে, কোডের অন্যত্র একশন প্রতিস্থাপন করে, একাধিক স্টেটমেন্টে লুপিং করে ইত্যাদি হচ্ছে এক্সেকিউটেবল স্টেটমেন্ট। আর একটি এসাইনমেন্ট স্টেটমেন্ট হলো এক্সেকিউটেবল স্টেটমেন্টের একটি বিশেষ রূপ। আমরা একটি উদাহরণ দেখতে পারি:

```
Module decisions
  Sub Main()
    'local variable definition '
    Dim a As Integer = 10

    ' check the boolean condition using if statement '
    If (a < 20) Then
      ' if condition is true then print the following '
      Console.WriteLine("a is less than 20")
    End If
    Console.WriteLine("value of a is : {0}", a)
    Console.ReadLine()
  End Sub
End Module
```

বায়ের কোড রান করে যে ফলাফল আসবে, তাহলো:

a is less than 20;
value of a is : 10

১৩. অপারেটর: Operators

গাণিতিক বা যৌক্তিক [লজিক্যাল] কোনো ক্রিয়া সম্পাদনের নির্দেশ দিতে ব্যবহৃত হয় কিছু প্রতীক। এগুলোকেই কম্পিউটার বিজ্ঞানে অপারেটর বলে। ভিজুয়্যাল বেসিক.নেটে প্রচুর ডিফাইন-করা অপারেটর আছে। নিচে তালিকাভুক্ত অপারেটরগুলো সর্বাধিক বেশি ব্যবহৃত হয়।

* পাটিগণিতের প্রতীক [Arithmetic Operators] যেমন: +, -, *, /, Mod, ^, \ [যোগ, বিয়োগ পূরণ ও ভাগ। Mod সর্বদাই ভাগশেষ দেবে। যেমন: ৭ Mod ২= ১। এটা [^] মানে পাওয়ার। সুতরাং ৭ ^ ২ = ৪৯। শেষোক্তটি শুধুমাত্র ইন্টেজার ফলাফল দেবে। যেমন: ৭ \ ২ = ৩]

* পার্থক্য নির্ণয়ক প্রতীক [Comparison Operators] যেমন: = [উভয় দিকের মান সমান], <> [উভয় দিকের মান সমান নয়], > [ডানদিকের মান বামদিক থেকে ছোট], < [ডানদিকের মান বাম দিক থেকে বড়ো], >= [বামদিকের মান ডানদিক থেকে বড়ো না হয় সমান], <= [বামদিকের মান ডানদিক থেকে ছোট বা সমান]

* লজিক্যাল / বিটওয়েজ প্রতীক [Logical/Bitwise Operators] যেমন: And, And এর উভয় দিকের মান সমান থাকলে ফলাফল True। Or, উভয় দিকের যে কোনো একটি সত্য হলে ফলাফল সত্য বা True। Not, কন্ডিশন সত্য হলে এটা একে মিথ্যায় পরিণত করে। উদাহরণ If Not A=10 then- অর্থাৎ A যদি ১০ না হয় তাহলে [সত্য]। ১০ হলে কিন্তু

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

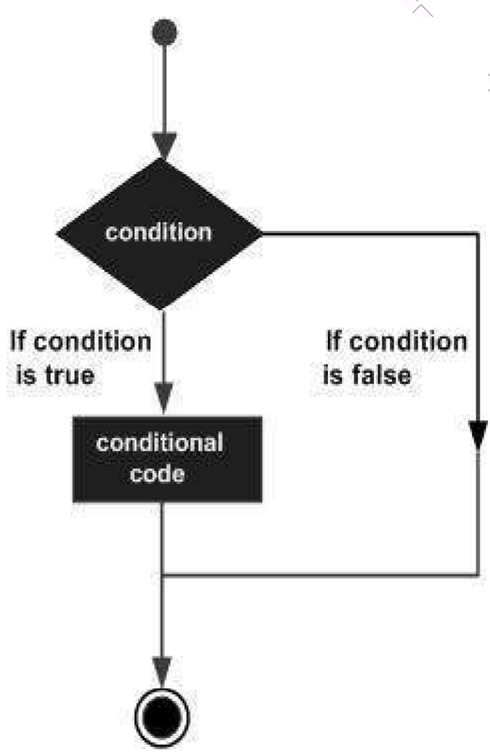
ফলাফল মিথ্যা হবে। Xor -উভয় এক্সপ্রেশন সত্য হলে ফলাফল সত্য [True]। অন্যথায় ফলাফল False। AndAlso- অর্থাৎ এবং যদিও। উভয় এক্সপ্রেশন সমান থাকলে ফলাফল True। OrElse - উভয়টির যে কোনো একটি সত্য হলে ফলাফল True। IsFalse -এটা বের করে কোনো একটি এক্সপ্রেশন মিথ্যা কি না। IsTrue -এটা নির্ধারণ করে কোনো একটি এক্সপ্রেশন সত্য কি না।

* বিট শিফ্‌ট অপারেটর [Bit Shift Operators] - এটা হচ্ছে বাইনারি পার্থক্য নির্ণয় পদ্ধতি। যেমন: A = 60 ও B = 13 হলে, বাইনারিতে তা হবে A = 0011 1100 এবং B = 0000 1101।

* এসাইনমেন্ট প্রতীক [Assignment Operators] -এগুলো হলো: =, += [Add AND- উদাহরণ: চ += ক, ক এর মান রেখে তাতে যোগ হবে চ এর মান। এটা চ = চ + ক এর সমান।]। -= [Subtract AND -আগেরটির বিপরীত ক্রিয়া]- চ = চ - ক এর সমান, । *= [Multiply AND - C *= A হলো C = C * A]। /= [Divide AND -C /= A হলো C = C/ A এর অনুরূপ]। \= [C \= A হলো C = C \A এর অনুরূপ]। ^= [C^=A হলো C = C ^ A এর অনুরূপ]।

১৪. সিদ্ধান্ত নেওয়া: Decision Making

প্রোগ্রামিংয়ে সিদ্ধান্ত নেওয়া একটি অবশ্যকরণীয় কাজ। ব্যবহারযোগ্য এমন কোনো এ্যাপলিকেশন হয়তো পাওয়া যাবে না যার কোডে সিদ্ধান্ত নেওয়ার এক বা একাধিক ব্লক নেই। টেস্টসহ সিদ্ধান্ত নেওয়া প্রোগ্রামিংয়ে একান্ত জরুরী একটি ব্যাপার। নিচের ফ্লো চার্টটি অধিকাংশ প্রোগ্রামিং ভাষায় মডেল হিসেবে ব্যবহৃত হয়।



ভিজুয়্যাল বেসিক.নেট ভাষার মধ্যে কিছু বিশেষ কিওয়ার্ড আছে যেগুলো ব্যবহৃত হয় সিদ্ধান্ত নিতে। যেমন:

১. If ... Then স্টেটমেন্টস্। ২. If ... Then .... Else স্টেটমেন্টস্ ৩. Nested If স্টেটমেন্টস্। ৪. Select Case স্টেটমেন্টস্। ৫. Nested Select Case স্টেটমেন্টস্। এগুলোর সব ক'টিই আমরা ব্যবহার করেছি এ গ্রন্থের বিভিন্ন উদহারণ এ্যাপলিকেশনে। এরপরও প্রতিটির ওপর একেকটি করে উদাহরণ দিচ্ছি সবার সুবিধার্থে।

১. a যদি ২০ থেকে কম বা সমান হয় তাহলে c= c+1 ।

```
If (a <= 20) Then
    c= c+1
End If
```

২. a যদি ২০ থেকে কম বা সমান হয় না তাহলে msgBox("a is larger then 20") ।

```
If (a <= 20) Then
    c= c+1
Else
    msgBox("a is larger then 20")
End If
```

৩. a যদি ১০০ হয় তাহলে b যদি ২০০ হয় তবেই পরের লাইন লিখা হবে: "Value of a is 100 and b is 200" ।

```
If (a = 100) Then
      ' if condition is true then check the following
      If (b = 200) Then
         ' if condition is true then print the following
         Console.WriteLine("Value of a is 100 and b is 200")
      End If
End If
```

১৮তম পরিচ্ছেদ: ***Visual Basic.Net: Simple Guide***

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

৪. নিচের কোডে গ্রেড A, B, C, D ও F পর্যন্ত সিলেক্ট কেইস ব্যবহৃত হয়েছে।

```
Sub Main()
    'local variable definition
    Dim grade As Char
    grade = "B"
    Select grade
        Case "A"
            Console.WriteLine("Excellent!")
        Case "B", "C"
            Console.WriteLine("Well done")
        Case "D"
            Console.WriteLine("You passed")
        Case "F"
            Console.WriteLine("Better try again")
        Case Else
            Console.WriteLine("Invalid grade")
    End Select
    Console.WriteLine("Your grade is  {0}", grade)
    Console.ReadLine()
End Sub
```

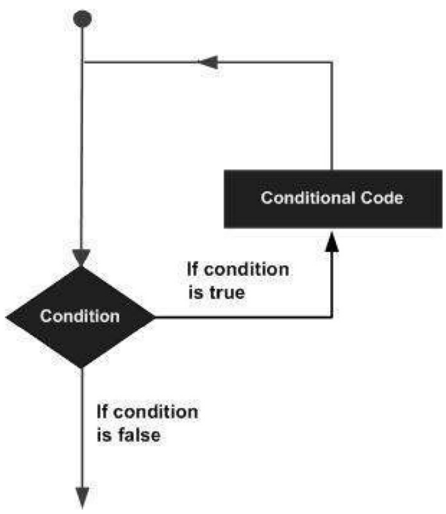
৫. নিচের কোডে নেস্টেড সিলেক্ট কেইস স্টেটমেন্ট ব্যবহৃত হয়েছে।

```
Module decisions
  Sub Main()
    'local variable definition
    Dim a As Integer = 100
    Dim b As Integer = 200
    Select a
        Case 100
            Console.WriteLine("This is part of outer case ")
            Select Case b
                Case 200
                    Console.WriteLine("This is part of inner case ")
            End Select
    End Select
    Console.WriteLine("Exact value of a is : {0}", a)
    Console.WriteLine("Exact value of b is : {0}", b)
    Console.ReadLine()
  End Sub
End Module
```

বায়ের প্রোগ্রাম রানের ফলাফল:
This is part of outer case
This is part of inner case
Exact value of a is : 100
Exact value of b is : 200

১৫. ভিজুয়্যাল বেসিক.নেট লুপ: VB.Net Loops

অধিকাংশ প্রোগ্রামিং ভাষায় লুপের লজিক ডানের চিত্রের মতো ব্যবহৃত হয়। প্রোগ্রামে লুপও একটি গুরুত্বপূর্ণ বিষয়। সুতরাং এ সম্পর্কে ভালো জ্ঞান থাকা আবশ্যক। সাধারণত, প্রোগ্রামের স্টেটমেন্ট একটার পর আরেকটি এক্সেকিউট হয়। সুতরাং কোথায় কোন্ কোড লাইন থাকবে তা-ও গুরুত্বহ। অপরদিকে লুপিংয়ের কাজ হলো কোনো শর্ত পূরণ না হওয়া পর্যন্ত লুপের ভেতর ঘুরে ঘুরে এক্সেকিউশন চলতে থাকবে। এ গ্রন্থের বিভিন্ন উদাহরণে অনেক লুপ আমরা দেখেছি।



১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সর্বমোট ৬ ধরনের লুপিং তার ভাষার মধ্যে সম্পৃক্ত করেছে। আপনার কাজের জন্য যেটি উপযোগী সেটিই ব্যবহার করবেন। নিচের টেবিলে এ ৬টি লুপ কাঠামোর ওপর তথ্যাদি তুলে ধরছি।

Loop Type	Description
Do Loop	It repeats the enclosed block of statements while a Boolean condition is True or until the condition becomes True. It could be terminated at any time with the Exit Do statement.
For...Next	It repeats a group of statements a specified number of times and a loop index counts the number of loop iterations as the loop executes.
For Each...Next	It repeats a group of statements for each element in a collection. This loop is used for accessing and manipulating all elements in an array or a VB.Net collection.
While... End While	It executes a series of statements as long as a given condition is True.
With... End With	It is not exactly a looping construct. It executes a series of statements that repeatedly refer to a single object or structure.
Nested loops	You can use one or more loops inside any another While, For or Do loop.

লুপ ব্যবহার পদ্ধতি: উপরের ৬টি লুপ কিভাবে কোডিং করতে হয় তার উদাহরণ এবার ততুলে ধরছি।

Do Loop

```
Module loops
  Sub Main()
    ' local variable definition
    Dim a As Integer = 10
    'do loop execution
    Do
       Console.WriteLine("value of a: {0}", a)
       a = a + 1
    Loop While (a < 20)
    Console.ReadLine()
  End Sub
End Module
======
Result:
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19
```

For...Next

```
Module loops
  Sub Main()
    Dim a As Byte
    ' for loop execution
    For a = 10 To 20
       Console.WriteLine("value of a: {0}", a)
    Next
    Console.ReadLine()
  End Sub
End Module
======
Result:
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19
value of a: 20
```

For Each...Next

```
Module loops
  Sub Main()
    Dim anArray() As Integer = {1, 3, 5, 7, 9}
    Dim arrayItem As Integer
   'displaying the values
    For Each arrayItem In anArray
       Console.WriteLine(arrayItem)
    Next
    Console.ReadLine()
  End Sub
End Module
======
Result:
1
3
5
7
9
```

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

While ... End While

```
Module loops
   Sub Main()
      Dim a As Integer = 10
      ' while loop execution '
      While a < 20
         Console.WriteLine("value of a:
{0}", a)
         a = a + 1
      End While
      Console.ReadLine()
   End Sub
End Module
======
Result:
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
value of a: 16
value of a: 17
value of a: 18
value of a: 19
```

Nested Loop

```
Module loops
  Sub Main()
    ' local variable definition
    Dim i, j As Integer
    For i = 2 To 100
      For j = 2 To i
        ' if factor found, not prime
        If ((i Mod j) = 0) Then
          Exit For
        End If
      Next j
      If (j > (i \ j)) Then
        Console.WriteLine("{0} is prime", i)
      End If
    Next i
    Console.ReadLine()
  End Sub
End Module
```

```
Result:
2 is prime
3 is prime
5 is prime
7 is prime
11 is prime
13 is prime
17 is prime
19 is prime
23 is prime
29 is prime
31 is prime
37 is prime
41 is prime
43 is prime
47 is prime
53 is prime
59 is prime
61 is prime
67 is prime
71 is prime
73 is prime
79 is prime
83 is prime
89 is prime
97 is prime
```

With...End With

```
Module loops
  Public Class Book
    Public Property Name As String
    Public Property Author As String
    Public Property Subject As String
  End Class
  Sub Main()
    Dim aBook As New Book
    With aBook
      .Name = "VB.Net Programming"
      .Author = "Engineer Azizul Bari"
      .Subject = "IT"
    End With
    With aBook
      Console.WriteLine(.Name)
      Console.WriteLine(.Author)
      Console.WriteLine(.Subject)
    End With
    Console.ReadLine()
  End Sub
End Module

Result:
VB.Net Programming
Engineer Azizul Bari
IT
```

এ ছ'টি প্রোগ্রাম ইচ্ছে হলে রান করে দেখতে পারেন। মনে রাখবেন লুপ কন্ট্রোল স্টেটমেন্ট কোড চলার সাধারণ পরম্পরাকে পরিবর্তন করে। লুপের ভেতর থেকে বের হওয়ার জন্য তিনটি কিওয়ার্ড ব্যবহার করা যায়। যেমন: ক. Exit - লুপ বা সিলেক্ট স্টেটমেন্ট থেকে বের হয়ে পরবর্তী কোডগুলো এক্সেকিউট হবে। খ. Continue - এটা বাকি কোড এক্সেকিউট না করে লুপের প্রথমে ফিরে যায়। গ. GoTo -এটা আজকাল তেমন বেশি ব্যবহৃত হয় না। কিওয়ার্ডের পরে একটি নাম থাকতে হবে। একে লেবেল বলে। এই নামটি যেখানে আছে সেখানে জাম্প মারবে।

```
Module loops
  Sub Main()
    ' local variable definition
    Dim a As Integer = 10
    ' while loop execution '
    While (a < 20)
      Console.WriteLine("value of a: {0}", a)
      a = a + 1
      If (a > 15) Then
        'terminate the loop using exit statement
        Exit While
      End If
    End While
    Console.ReadLine()
  End Sub
End Module
```

বায়ের কোড রানের ফলাফল:

```
value of a: 10
value of a: 11
value of a: 12
value of a: 13
value of a: 14
value of a: 15
```

লক্ষ করুন, Exit While লাইনটি কার্যকর হবে যেই মুহূর্তে a এর মান ১৫ থেকে বেশি তথা ১৬ হবে।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

১৫. ভিজুয়্যাল বেসিক.নেটে স্ট্রিং নিয়ে কাজ: VB.Net Strings Manipulation

ভিজুয়্যাল বেসিক.নেটে String শব্দটি বিশেষ স্থান দখল করে আছে। স্ট্রিং মানে কিবোর্ডের যে কোনো এক বা একাধিক ক্যারাক্টারকে বুঝায়। String একটি টাইপও বটে। যেমন: আমরা বলতে পারি Dim Name as String। এই ডিক্লারেশনে String টাইপ। স্ট্রিং নিয়ে অনেক কাজ করা যায়। নিচে কছু উদাহরণ দেওয়া হলো।

১. স্ট্রিং বস্তু সৃষ্টির পদ্ধতি ৫টি। যথা: (ক) স্ট্রিং ভেরিয়েবলে স্ট্রিং লিটারেল এসাইন করা। যেমন: Dim Name as String; Name = "E. A. Bari"। (খ) স্ট্রিং কন্সট্রাক্টারের ব্যবহার। যেমন: Dim letters As Char() = {"H", "e", "l", "l", "o"}; greetings = New String(letters)। (গ) স্ট্রিং যোগ [+] অপরেটর ব্যবহার। যেমন: Dim fName, sName, FullName as String; fName = "Muhammad"; sName = "Hassan"; FulName = fName + " " + sName। (ঘ) মেথড কল করে স্ট্রিং রিটার্ন করা। যেমন: Dim sarray() As String = {"Hello", "From", "Subidbazar", "Point."}; Dim message As String = String.Join(" ", sarray)। এখন message = Hello From Subidbazar Point.। (ঙ) মেথড ফরম্যাট করে। যেমন: Dim waiting As DateTime = New DateTime(2017, 1, 12, 17, 58, 1); Dim chat As String = String.Format("Message sent at {0:t} on {0:D}", waiting); Console.WriteLine("Message: {0}", chat)। ফলাফল: Message: Message sent at 5:58 PM on Thursday, January 12, 2017 ।

স্ট্রিং অবেক্টের অনেক ফাংশন ও মেথড আছে। ওগুলোর ওপর এখানে আর কিছু বলে কলেবর বৃদ্ধি করছি না। শুধুমাত্র নামগুলো এখানে উল্লেখ করবো। প্রোগ্রাম করাকালে আপনি ইন্টারনেট বা স্টুডিও এর Help থেকে সহজেই জানতে ও ব্যবহার করতে পারবেন।

1. String.Compare(str1, str2) [result 0 if both are equal]
2. str1.contains(str2) - if str1 contains str2 then result is True, otherwise False
3. string.Substring (position as integer)gets the sring's position to end. ex. "This Book".Substring(5) -result Book.
4. String.Join (vbCrLf, Sring Value) - joins string.

Example:

```
Dim strarray As String() = {"Down the way where the nights are grey",
                 "And the sun shines daily on the mountain top",
                  "I took a trip on a sailing ship",
                 "And when I reached Jamaica",
                 "I made a stop"}
Dim str As String = String.Join(vbCrLf, strarray)
Result:   Down the way where the nights are grey
          And the sun shines daily on the mountain top
          I took a trip on a sailing ship
          And when I reached Jamaica
          I made a stop
```

১৬. তারিখ নিয়ে কাজ: Working with Date

ভিজুয়্যাল বেসিক.নেট বেশ কয়েকটি Date ফাংশন তৈরি করেছে আমাদের সুবিধার্থে। এর ক'টির ব্যবহার ডানের কোড অংশে দেখুন। ফলাফল নিম্নরূপ।

12/16/2012 12:00:00 PM
12/16/2012 12:00:52 PM
1/18/2017 9:55:18 PM [Local Time BD]
1/18/2017 3:55:18 PM [GMT]
1/18/2017 12:00:00 AM
Current Time: 9:55:18 PM

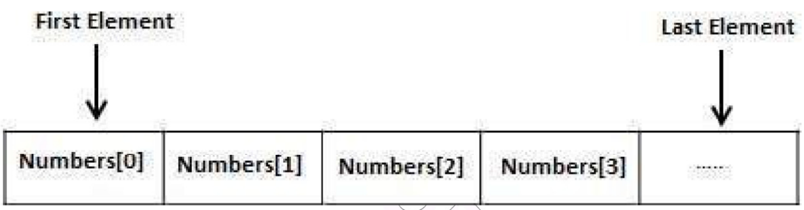
```
Module Module1
  Sub Main()
    'DateTime constructor: parameters year, month,
day, hour, min, sec
    Dim date1 As New Date(2012, 12, 16, 12, 0, 0)
    'initializes a new DateTime value
    Dim date2 As Date = #12/16/2012 12:00:52 AM#
    'using properties
    Dim date3 As Date = Date.Now
    Dim date4 As Date = Date.UtcNow
    Dim date5 As Date = Date.Today
    Console.WriteLine(date1)
    Console.WriteLine(date2)
    Console.WriteLine(date3)
    Console.WriteLine(date4)
    Console.WriteLine(date5)
    Console.Write("Current Time: ")
    Console.WriteLine(Now.ToLongTimeString)
    Console.ReadKey()
  End Sub
End Module
```

১৮তম পরিচ্ছেদ: ***Visual Basic.Net: Simple Guide***

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

১৭. ভিজুয়্যাল বেসিক.নেট এ্যারে: VB.Net - Arrays

এ্যারে ভেরিয়েবলে তথ্য গুচ্ছ একত্রে সংরক্ষণ করা যায়। এ্যারের মধ্যে থাকে ০-ইনডেক্স ইলেমেন্ট। নিচের চকটি দেখুন।



আমরা Dim সেস্টমেন্ট দ্বারা এ্যারে সৃষ্টি করি। যেমন:

```
Dim intData(30)  ' an array of 31 elements
Dim strData(20) As String' an array of 21 strings
Dim twoDarray(10, 20) As Integer 'a two
dimensional array of integers
Dim ranges(10, 100) 'two dimensional array
```

ডিক্লারেশনের সময় এ্যারে ইনিশিয়েলিজও করা যায়। যেমন:

```
Dim intData() As Integer = {12, 16, 20, 24, 28, 32} 'contains 0-5 indexed integer array elements
Dim Mnames() As String = {"Boishak", "Joysta", _
"Ashar", "Srabon", "Vadra"} 'contains 0-4 indexed string array elements
Dim miscData() As Object = {"Hello World", 12d, 16ui, "A"c} 'contains 0-3 indexed mixed types array elements
```

যে কোনো এ্যারে ভেরিয়েবলে সংরক্ষিত ডাটা ইলেমেন্ট ইনডেক্সট ব্যবহার করে বের করে আনা যায়। যেমন উপরের প্রথম এ্যারের চতুর্থ সংখ্যাটি পেতে আমরা লিখবো: intData(3) ফলাফল হবে ২৪। অনুরূপ: Mnames(0) এর ফলাফল হবে Boishak।

ডাইনামিক এ্যারে: Dynamic Arrays: এর অর্থ প্রোগ্রাম চলাকালে প্রয়োজনে আমরা এ্যারে ডাইমেনশন করতে পারি ও পুনঃডাইমেনশনও করতে পারি। এজন্য আমরা ব্যবহার করি ReDim কিওয়ার্ডটি। এটা ঘোষণভর Syntax হলো:
ReDim [Preserve] arrayname(subscripts)

লক্ষ করুন, অপশন্যাল কিওয়ার্ড Preserve ব্যবহার করলে এ্যারের পূর্বের ইলেমেনন্টগুলোও অপরিবর্তিত থাকবে। যেমন:

```
Dim ExamMarks() as Integer 'array declaration
ReDim ExamMarks(5) 'now the array has 0-6 indexed integer elements.
ReDim Preserve ExamMarks(10) 'now the array has 0-11 indexed integer elements.
```

একাধিক বিস্তৃতির এ্যারে: Multi-Dimensional Arrays: এ্যারে ভেরিয়েবল একাধিক বিস্তৃতিসম্পন্নও হতে পারে। যেমন আমরা ডিক্লায়ার করতে পারি: DIm twoDStringArray(10, 20) As String। এটা ২-বিস্তৃতির এ্যারে। ৩-বিস্তৃতিসম্পন্ন এ্যারে ডিক্লারেশন হলো: Dim threeDIntArray(10, 10, 10) As Integer। নিচে ২-বিস্তৃতির একটি এ্যারে কোড ব্লক দেখুন।

```
Module arrayApl
  Sub Main()
    ' an array with 5 rows and 2 columns
    Dim a(,) As Integer = {{0, 0}, {1, 2}, {2, 4}, {3, 6}, {4, 8}}
    Dim i, j As Integer
    ' output each array element's value '
    For i = 0 To 4
      For j = 0 To 1
        Console.WriteLine("a[{0},{1}] = {2}", i, j, a(i, j))
      Next j
    Next i
    Console.ReadKey()
  End Sub
End Module
```

বায়ের কোড থেকে প্রাপ্ত ফলাফল:

```
a[0,0]: 0
a[0,1]: 0
a[1,0]: 1
a[1,1]: 2
a[2,0]: 2
a[2,1]: 4
a[3,0]: 3
a[3,1]: 6
a[4,0]: 4
a[4,1]: 8
```

ভিজুয়্যাল বেসিক.নেট এ্যারে ক্লাসের বেশ ক'টি মেথড তৈরি করে দিয়েছে। আমরা এগুলো প্রয়োজনে কাজে লাগাতে

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

পারি। এসব মেথডের মধ্যে উল্লেখযোগ্য ক'টি হলো:
Clear, Copy, CopyTo, GetLength, GetLongLength, GetLowerBound, GetType, GetUpperBound, GetValue, IndexOf, Reverse, SetValue, Sort, ToString.

এবার একটা উদাহরণ দেখে নিই। নিচের মডিউলে এ্যারে ক্লাসের কিছু মেথড ব্যবহারের নিয়ম তুলে ধরা হয়েছে।

```
Module arrayApl
   Sub Main()
      Dim list As Integer() = {34, 72, 13, 44, 25, 30, 10}
      Dim temp As Integer() = list
      Dim i As Integer
      Console.Write("Original Array: ")
      For Each i In list
         Console.Write("{0} ", i)
      Next i
      Console.WriteLine()
      ' reverse the array
      Array.Reverse(temp)
      Console.Write("Reversed Array: ")
      For Each i In temp
         Console.Write("{0} ", i)
      Next i
      Console.WriteLine()
      'sort the array
      Array.Sort(list)
      Console.Write("Sorted Array: ")
      For Each i In list
         Console.Write("{0} ", i)
      Next i
      Console.WriteLine()
      Console.ReadKey()
   End Sub
End Module
```

বায়ের প্রোগ্রাম কম্পাইল করে রান করলে যে ফলাফল পাওয়া যাবে তাহলো:
Original Array: 34 72 13 44 25 30 10
Reversed Array: 10 30 25 44 13 72 34
Sorted Array: 10 13 25 30 34 44 72

লক্ষ করুন, উপরের প্রথম লিস্ট হলো মূল এ্যারে। দ্বিতীয়টি উল্টো এ্যারে এবং তৃতীয়টি ছোট থেকে বড় সর্টকরা এ্যারে।

১৮. ফাংশন: VB.Net - Functions

ফাংশনের ওপর বেশ কিছু আলোচনা, ব্যাখ্যা ও ব্যবহার উদাহরণ আমরা এই গ্রন্থে তুলে ধরেছি। এরপরও আরো কিছু জানার আছে। ফাংশন মূলত সাবরুটিন প্রসিডোর এর মতো কাজ করে। তবে ফাংশন থেকে আমরা পেতে পারি মান- যা এটা কলিং সাবে রিটার্ন করে। সুতরাং বলা যায়, ফাংশন একটি ভ্যালু রিটার্ন করে কিন্তু সাবরুটিন তা করে না।

ফাংশন ডিফাইন করা নিয়ম হলো:

```
[Modifiers] Function FunctionName [(ParameterList)] As ReturnType
   [Statements]
End Function
```

এখানে, Modifiers অপশন্যাল এবং তা নির্দিষ্ট করে ফাংশন প্রাপ্তির লেবেল। এগুলো হলো: Public, Private, Protected, Friend, Protected Friend । FunctionName জরুরি। ParameterList অপশন্যাল। সবশেষে ReturnType নির্দিষ্ট করে কোন্ ধরনের ডাটা ফাংশন রিটার্ন করবে।

ফাংশনের বিভিন্ন দিক বুঝতে আমরা কয়েকটি উদাহরণ দেখতে পারি। পরের পৃষ্ঠায় এসব উদাহরণ তুলে ধরেছি। শিক্ষার্থীরা চাইলে এসব ফাংশনের ফলাফল প্রজেক্ট তৈরি করে পরীক্ষা করতে পারেন। বাস্তবে তা করাটাই উত্তম।

১৮তম পরিচ্ছেদ: ***Visual Basic.Net: Simple Guide***

ভিজুয়াল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

ফাংশন উদাহরণ -১: এটা দু'টি ইন্টেজার প্যারামিটার নেয় এবং রিটার্ন করে এদের মধ্যে বড়ো সংখ্যাটি।

```
Function FindMax(ByVal num1 As Integer, ByVal num2 As Integer) As Integer
   ' local variable declaration */
   Dim result As Integer
   If (num1 > num2) Then
       result = num1
   Else
       result = num2
   End If
   FindMax = result
End Function
```

উদাহরণ -২: ভিজুয়াল বেসিক.নেটে ফাংশন ভ্যালু বা মান রিটার্ন করে দু'টি পদ্ধতিতে। প্রথমটি হলো Return কিওয়ার্ড দ্বারা আর দ্বিতীয়টি হলো ফাংশনের নামের দ্বারা। উপরের ফংশনে ব্যবহৃত হয়েছে দ্বিতীয় পদ্ধতি। নিচের ফাংশন প্রথম পদ্ধতিতে ভ্যালু রিটার্ন করবে। এটা সংখ্যার ফ্যাক্টোরিয়েল মান বের করে।

```
Module myfunctions
   Function factorial(ByVal num As Integer) As Integer
      ' local variable declaration */
      Dim result As Integer
      If (num = 1) Then
          Return 1
      Else
          result = factorial(num - 1) * num
          Return result
      End If
   End Function
```

উদাহরণ - ৩: কেনো কোনো সময় এমন হয় যে, ফাংশনের মধ্যে ক'টি প্যারামিটার হবে তা নির্দিষ্ট করে বলা যায় না। এমতাবস্থায় কি করবো? ভিজুয়াল বেসিক.নেট এরূপ ক্ষেত্রে একটি কিওয়ার্ড দিয়েছে। এর নাম ParamArray। ফাংশন ডিক্লারেশনে এটি ব্যবহার করে অনির্দিষ্ট ইলেমেন্টবিশিষ্ট এ্যারে ব্যবহার করতে পারি ফাংশনের প্যারামিটার হিসাবে। দেখুন এই উদাহরণটি।

```
Module myparamfunc
   Function AddElements(ParamArray arr As Integer()) As Integer
      Dim sum As Integer = 0
      Dim i As Integer = 0
      For Each i In arr
          sum += i
      Next i
      Return sum
   End Function
   Sub Main()
      Dim sum As Integer
      sum = AddElements(512, 720, 250, 567, 889)
      Console.WriteLine("The sum is: {0}", sum)
      Console.ReadLine()
   End Sub
End Module
```

ফাংশনের ব্যাখ্যা: আমরা এখানে ParamArray কিওয়ার্ড দ্বারা একটি অনির্দিষ্ট ইলেমেন্টবিশিষ্ট প্যারামিটার তৈরি করেছি। ঠিক ক'টি ইলেমেন্ট থাকবে তা কলিং সাব দ্বারা জানিয়েছি। বলেছি: sum = AddElements(512, 720, 250, 567, 889)। সুতরাং প্যারামিটার হয়েছে মোট ৫টি।

ফাংশনের ফলাফল হলো এই ৫টি সংখ্যার যোগফল:

The sum is: 2938

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

উদাহরণ -৪: সময় সময় এ্যারেকে ফাংশনের আর্গুমেন্ট হিসেবেও পাস করা যায়। যেমন:

```
Module arrayParameter
  Function getAverage(ByVal arr As Integer(), ByVal size As Integer) As Double
    'local variables
     Dim i As Integer
     Dim avg As Double
     Dim sum As Integer = 0
     For i = 0 To size - 1
        sum += arr(i)
     Next i
     avg = sum / size
     Return avg
  End Function
  Sub Main()
     ' an int array with 5 elements '
     Dim balance As Integer() = {1000, 2, 3, 17, 50}
     Dim avg As Double
     'pass pointer to the array as an argument
     avg = getAverage(balance, 5)
     ' output the returned value '
     Console.WriteLine("Average value is: {0} ", avg)
     Console.ReadLine()
  End Sub
End Module
```

ফাংশনের ব্যাখ্যা: ব্যবহারকারী ক'টি সংখ্যার গড় বের করবেন তা আমাদের জানা নেই। সুতরাং আমরা একটি ফাংশন তৈরি করলাম যা মূলত ফ্লেক্সিবল। বায়ের কলিং সাবে আমরা বলেছি balance নামক Integer এ্যারেতে মোট ৫টি সংখ্যা আছে। আমরা এগুলোর গড় ফাংশন থেকে বের করার নির্দেশ দিয়েছি। লক্ষ করুন নির্দেশটি হলো এই লাইন:

avg = getAverage(balance, 5)

ফাংশনের প্রথম প্যারামিটার একটি অনির্দিষ্ট এ্যারে। আর দ্বিতীয়টি এ্যারের সাইজ। উপরের লাইন দ্বারা এ্যারে ও সাইজ উভয়টি নির্দিষ্ট করে ফাংশন কল করেছি। আর সঠিক ফালাফলও পেয়েছি:

Average value is: 214.4

১৮. সাব প্রসিডোর: VB.Net - Sub Procedure

এই গ্রন্থের সর্বত্র আমরা সাব প্রসিডোর ব্যবহার করে আসছি। সুতরাং প্রসিডোর সম্পর্কে শিক্ষাথীরা পূর্ণাঙ্গরূপে অবগত আছেন। এরপরও দু'একটি বিষয় আমাদের জানার আছে। এখন ওসব ব্যাপার নিয়ে সামান্য আলোচনা করবো। প্রথমে দেখে নিই সাব প্রসিডোরের সিনট্যাক্সট।

```
[Modifiers] Sub SubName [(ParameterList)]
    [Statements]
End Sub
```

অপশন্যাল Modifiers হতে পারে: Public, Private, Protected, Friend, Protected Friend । একটি উদাহরণ দেখে নিই।

```
Module mysub
  Sub CalculatePay(ByRef hours As Double, ByRef wage As Decimal)
    'local variable declaration
    Dim pay As Double
    pay = hours * wage
    Console.WriteLine("Total Pay: {0:C}", pay)
  End Sub
  Sub Main()
    'calling the CalculatePay Sub Procedure
    CalculatePay(25, 10)
    CalculatePay(40, 20)
    CalculatePay(30, 27.5)
    Console.ReadLine()
  End Sub
End Module
```

সাবের ব্যাখ্য ও ফলাফল: CalculatePay সাবরুটিনে আছে hours এবং wage নামক দু'টি প্যারামিটার। সাবই ফলাফল দিয়েছে। কলিং নাম [Sub Main] থেকে আমরা তিনবার সাবরুটিন কল করেছি। আর ফলাফল যা পেয়েছি তাহলো:

Total Pay: $250.00
Total Pay: $800.00
Total Pay: $825.00

ভিজুয়্যাল বেসিক.নেটে ফাংশন ও সাব প্রসিডোরের আর্গুমেন্ট বা প্যারামিটার দু'ভাবে ডিফাইন করা যায়। এগুলো হলো: ByValue এবং ByRef। প্রথমটি হলো ডিফোল্ট। প্রথমটির অর্থ আর্গুমেন্ট অপরিবর্তিত থাকবে, দ্বিতীয়টির অর্থ, প্রয়োজনে ফাংশন বা সাবরুটিন আর্গুমেন্টের মান রদবদল করতে পারবে।

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

১৯. ক্লাস ও অবজেক্ট: Classes & Objects

ক্লাস ডেফিনিশন মানে একটি ডাটা টাইপের জন্য ব্লুপ্রিন্ট তৈরি করা। অবজেক্ট হলো ক্লাসের ইন্সট্যান্ট। ক্লাসের ভেরতে যেসব মেথড ও ভেরিয়েবল থাকে ওগুলোকে বলে ক্লাসের মেমবার। ক্লাসের একটি উদাহরণ দেখে নিই।

```
Class Box
    Public length As Double   ' Length of a box
    Public breadth As Double   ' Breadth of a box
    Public height As Double   ' Height of a box
End Class
```

উপরের ক্লাসে একটি বাক্সের ডেফিনিশন এসছে। এর ভেতর তিনটি সদস্য আছে: length, breadth and height । এটাকে কাজে লাগাতে আমরা প্রথমে ইন্সট্যান্ট ডিক্লারেশন করে থাকি। যথা:

```
Dim Box1 As Box = New Box()      ' Declare Box1 of type Box
Dim Box2 As Box = New Box()      ' Declare Box2 of type Box
```

এখন আমরা লিখতে পারি:

```
Box1.height = 5.0
Box1.length = 6.0
Box1.breadth = 7.0
' box 2 specification
Box2.height = 10.0
Box2.length = 12.0
Box2.breadth = 13.0
```

উপরোক্ত ভ্যালুগুলো নিয়ে ক্যালকুলেশনের জন্য ডিক্লায়ার করতে পারি আরেকটি ভেরিয়েবল। যেমন:

```
Dim volume As Double = 0.0
```

চাইলে আমরা ক্লাসের ভেতরই উক্ত বাক্সের তিনটি ডাইমেনশন ডিফাইন ও ক্যালকুলেট করতে পারি। যেমন:

```
Class Box
    Public length As Double   ' Length of a box
    Public breadth As Double   ' Breadth of a box
    Public height As Double   ' Height of a box
    Public Sub setLength(ByVal len As Double)
       length = len
    End Sub
    Public Sub setBreadth(ByVal bre As Double)
       breadth = bre
    End Sub
    Public Sub setHeight(ByVal hei As Double)
       height = hei
    End Sub
    Public Function getVolume() As Double
       Return length * breadth * height
    End Function
  End Class
```

লক্ষ করুন, আমরা প্রথম ক্লাস ডেফিনিশনে শুধুমাত্র বাক্সের তিনটি ডাইমেনশনের নাম টাইপসহ ডিক্লায়ার করেছি। কিন্তু উপরের ক্লাসে তিনটি ডাইমেনশন সেট ও ভলিউম বের করার ফাংশনও ডিফাইন করেছি। এর অর্থ হলো কলিং সাবের জন্য কাজটি সহজ করে দিলাম। এখন কলিং সাবে আমরা লিখতে পরি:

```
Dim Box1 As Box = New Box()      ' Declare Box1 of type Box
Dim volume As Double = 0.0    ' Store the volume of a box here
```

এরপর আমরা সেট করতে পারি মান: [পরের পৃষ্ঠায় দেখুন]

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

```
' box 1 specification
Box1.setLength(6.0)
Box1.setBreadth(7.0)
Box1.setHeight(5.0)
' volume of box 1
volume = Box1.getVolume() 'function call within the Box class
```

বর্ণিত উপায়ে ক্লাস ডিক্লারেশনকে বলে Member Functions and Encapsulation পদ্ধতি।

২০. এক্সসেপশন বা এ্যারোর নিয়ন্ত্রণ: Exception or Error Handling

কোডিং করতে যেয়ে ভুল হবেই। সামান্য ভুলের কারণে পুরো এ্যাপলিকেশনই নষ্ট হয়ে যেতে পারে। এমনকি রানকালে কম্পিউটার ক্রাশও করতে পারে। অনাকাঙ্ক্ষিত এই অবস্থা থেকে নিষ্কৃতির কিছু উপায় ভিজুয়্যাল বেসিক.নেট তৈরি করেছে। এসব উপায় অবলম্বনের নামই হলো **Exception or Error Handling**।

সাধারণত কোডের যে অংশে ভুল আছে সেখান থেকে প্রোগ্রাম ফ্লো অন্যত্র ট্রান্সফার করে নেওয়া হয়ে থাকে। ভিজুয়্যাল বেসিক.নেট ভুল সংশোধনের জন্য চারটি কিওয়ার্ড দিয়েছে: **Try, Catch, Finally** and **Throw**। আরেকটি কিওয়ার্ড আছে। আজকাল এটি খুব বেশি ব্যবহৃত হয় না। যদিও এক সময় এটা খুব জনপ্রিয় ছিলো। এই কিওয়ার্ডটি হলো: GoTo [label]। অর্থাৎ নির্দিষ্ট লেবেলে জাম্প করার নির্দেশ। যা হোক প্রথমোক্ত চারটি কিওয়ার্ড কিভাবে কাজ করে একটু দেখে নিই।

```
Module exceptionProg
  Sub division(ByVal num1 As Integer, ByVal num2 As Integer)
    Dim result As Integer
    Try
      result = num1 \ num2
    Catch e As DivideByZeroException
      Console.WriteLine("Exception caught: {0}", e)
    Finally
      Console.WriteLine("Result: {0}", result)
    End Try
  End Sub
  Sub Main()
    division(25, 0)
    Console.ReadKey()
  End Sub
End Module
```

বায়ের সাব ও ফলাফল: এখানে Try ... End Try ব্লকের ভেতর ভুল ধরার ব্যবস্থা করা হয়েছে। মনে করুন ০ দ্বারা ভাগের চেষ্টা করা হলো, যেমনটি Sub Main-এ করা হয়েছে। যেহেতু এটা ভুল তাই উক্ত ব্লকে এটা ধরা পড়বে ও প্রোগ্রাম রিপোর্ট দেবে। বায়ের কোড রানের ফলাফল হবে:

Exception caught: System.DivideByZeroException: Attempted to divide by zero.
at ...
Result: 0

প্রোগ্রামার নিজেও ভুল ধরার ডেফিনিশন তৈরি করতে পারেন। যেমন:

```
Public Class TempIsZeroException : Inherits ApplicationException
    Public Sub New(ByVal message As String)
      MyBase.New(message)
    End Sub
End Class
Public Class Temperature
    Dim temperature As Integer = 0
    Sub showTemp()
      If (temperature = 0) Then
        Throw (New TempIsZeroException("Zero Temperature found"))
      Else
        Console.WriteLine("Temperature: {0}", temperature)
      End If
    End Sub
End Class
```

বায়ের প্রথম ক্লাস ডিক্লারেশনের সময় ApplicationException নামক সিস্টেম ক্লাস ব্যবহৃত হয়েছে। এরপর ক্লাসে কিওয়ার্ড দ্বারা এক্সেস এক্সপ্রেশনের মেসেজটি ডিফাইন করা হয়েছে। সুতরাং ভুল হলে লেখা আসবে: TempIsZeroException: Zero Temperature found

১৮তম পরিচ্ছেদ: *Visual Basic.Net: Simple Guide*

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

২১. ফাইল নিয়ে কাজ: File Handling

ফাইল কি? ফাইল হচ্ছে একগুচ্ছ তথ্য যা ডিস্কের মধ্যে সংরক্ষিত থাকে। ফাইল থাকে ডাইরেক্টরিতে যার একটি নির্দিষ্ট প্যাথ বা রাস্তা আছে। যে মুহূর্তে কোনো ফাইল খুলবে পড়া ও লেখার জন্য, তখনই এটা হয়ে যায় ফাইল-স্ট্রীম। স্ট্রীম কি? এটা মূলত পরম্পরা তথ্যের বাইটস যোগাযোগ রাস্তা ধরে চলমান থাকে। দু'টি মূল স্ট্রীম আছে: ইনপুট [প্রবেশকরণ] ও আউটপুট [বেরকরণ]। ইনপুট স্ট্রিম ব্যবহৃত হয় ফাইল থেকে ডাটা পড়ার জন্য। আর আউটপুট স্ট্রিম দ্বারা ফাইলের মধ্যে ডাটা লিখা হয়।

ভিজুয়্যাল বেসিক.নেট সিস্টেম আইও [System.IO namespace] নেইমস্পেস তৈরি করেছে ফাইল নিয়ে কাজ করার জন্য। যেমন: ফাইল সৃষ্টি, ধ্বংস, খোলা, বন্ধ করা, পড়া, লেখা, সংরক্ষণ করা ইত্যাদি। নিচের টেবিলে System.IO namespace-এর সচরাচর ব্যবহৃত কয়েকটি ক্লাস ব্যাখ্যাসহ তুলে ধরেছি। পরে কিছু উদাহরণ কোডও লিপিবদ্ধ করেছি।

I/O Class	Description
BinaryReader	Reads primitive data from a binary stream.
BinaryWriter	Writes primitive data in binary format.
BufferedStream	A temporary storage for a stream of bytes.
Directory	Helps in manipulating a directory structure.
DirectoryInfo	Used for performing operations on directories.
DriveInfo	Provides information for the drives.
File	Helps in manipulating files.
FileInfo	Used for performing operations on files.
FileStream	Used to read from and write to any location in a file.
MemoryStream	Used for random access of streamed data stored in memory.
Path	Performs operations on path information.
StreamReader	Used for reading characters from a byte stream.
StreamWriter	Is used for writing characters to a stream.
StringReader	Is used for reading from a string buffer.
StringWriter	Is used for writing into a string buffer.

১. ফাইলস্ট্রিম ক্লাস: FileStream Class

এ ক্লাসের মাধ্যমে আমরা নির্দিষ্ট ফাইল খুলে পড়া, লেখা ও বন্ধ করার মতো কাজ করতে পারি। একটি উদাহরণ থেকে এর ব্যবহার স্পষ্ট হবে।

এটি একটি অতি সাধারণ উদাহরণ। ফাইল নিয়ে কাজ করার একাধিক শক্তিশঅলী ক্লাস আছে। পরের পৃষ্ঠা থেকে বেশ ক'টি উদাহরণ শিক্ষার্থীদের সুবিধার্থে উপস্থাপন করেছি। খুব ভালো করে ওগুলো স্টাডি করবেন। ফাইল নিয়ে কাজ করা প্রোগ্রামিংয়ের সঙ্গে অঙ্গাঙ্গিভাবে জড়িত। সুতরাং একে জানা একান্ত জরুরী।

```
Imports System.IO
Module fileProg
  Sub Main()
    Dim f1 As FileStream = New FileStream("sample.txt", _
         FileMode.OpenOrCreate, FileAccess.ReadWrite) _
    Dim i As Integer
    For i = 0 To 20
       f1.WriteByte(CByte(i))
    Next i
    f1.Position = 0
    For i = 0 To 20
       Console.Write("{0} ", f1.ReadByte())
    Next i
    f1.Close()
    Console.ReadKey()
  End Sub
End Module
```

১৮তম পরিচ্ছেদ: ***Visual Basic.Net: Simple Guide***

ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

ক. টেস্টট্ ফাইল থেকে পড়া ও লেখা: এ জন্য মাত্র ক'টি কোড লাইনই যথেষ্ট। যেমন:

```
Private Sub ReadTxtFile
    Dim R as New IO.StreamReader("") 'put the full file path between the quote marks
    txtText.Text = R.ReadToEnd
    R.Close()
End Sub
```

উক্ত কোড প্যাথ দ্বারা চিহ্নিত টেক্সট্ ফাইল খুলে টেক্সট্ বাক্সে দেখাবে। স্ট্রিমরিডার ক্লাসকে ফাইল পাঠের পর বন্ধ করতে হবে। সুতরাং তৃতীয় লাইনটি এখানে জরুরী।

খ. টেস্টট্ ফাইল থেকে পড়া, এডিট করা ও সেইভ করা: এ কাজগুলোও তেমন বেশি কঠিন নয়। আপনার সলুশনে যুক্ত করুন একটি OpenFileDialog ও একটি SaveFileDialog অবজেক্ট। এরপর নিচের কোডটুকু লিখুন। যথাক্রমে অপেন ও ক্লোজ বোতাম থেকে সাবগুলো কল করুন।

```
imports System.IO 'এটা উপরে ডিক্লায়ার হবে।
Private Sub ReadWriteTxtFile
    Dim FileReader as StreamReader
    Dim Results as DialogReasult
    Resutls = OpenFileDialog1.ShowDialog
    If Results = DialogReasult.OK then
        FileReader = New StreamReader(OpenFileDialog1.FileName)
        txtText1.Text = FileReader.ReadToEnd()
        FileReader.Close()
    End If
End Sub

Private Sub ReadWriteTxtFile
    Dim FileWriter as StreamWriter
    Dim Results as DialogReasult
    Resutls = SaveFileDialog1.ShowDialog
    If Results = DialogReasult.OK then
        FileWriter = New StreamWriter(SaveFileDialog1.FileName, False)
        FileWriter.Write(txtText.Text)
        FileWriter.Close()
    End If
End Sub
```

১৮তম পরিচ্ছেদ: ***Visual Basic.Net: Simple Guide***

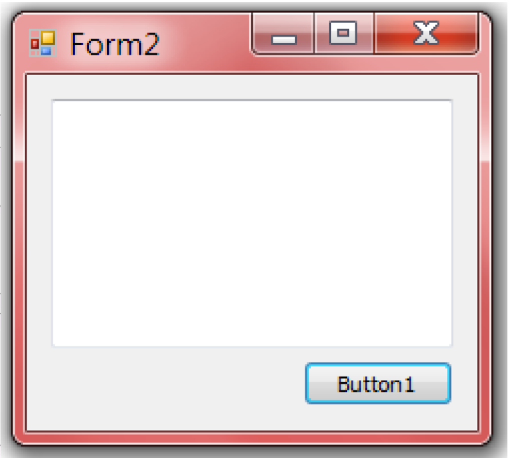
ভিজুয়্যাল বেসিক.নেট সহজ গাইড: *Visual Basic.Net: Simple Guide*

২২. ফর্মে কন্ট্রোল ডক ও নঙ্গর করা: Docking and Anchoring Controls on the Form

ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিংয়ের ওপর রচিত এ গ্রন্থের শেষ পর্যায়ে এসে গেছি আমরা! বাস্তবে জানার তো শেষ নেই। কিন্তু এক পর্যায়ে তো গ্রন্থের ইতি টানতেই হবে- তাই না? আমি আশাকরি শিক্ষার্থীরা এ গ্রন্থ পাঠ করে ও এতে সন্নিবেশিত বিভিন্ন উদাহরণ এ্যাপলিকেশন থেকে উপকৃত হয়েছেন। ভিজুয়্যাল বেসিক.নেট একটি শক্তিশালী উইন্ডোজ প্রোগ্রামিং ভাষা। শিক্ষার্থীদের প্রতি আমার উপদেশ হচ্ছে, ইন্টারনেট থাকলে [আর আজকাল কার ইনন্টারনেট নেই!] ভিজুয়্যাল বেসিক.নেট প্রোগ্রামিং শেখার অসংখ্য ওয়েভসাইট থেকে উপকৃত হতে পারেন। আপনি যা কিছুই করতে চান না কেনো, সেসব বিষয়ের ওপর উদাহরণ কোড ও প্রজেক্ট অবশ্যই খুঁজে পাবেন। সময় সময় কিছুটা অতিরিক্ত ধৈর্যশীল হতে হবে- এই যা।

যাক, আমরা একটি গুরুত্বপূর্ণ বিষয়ের ওপর কিঞ্চিৎ আলোচনা করে এ রচনার ইতি টানছি। বিষয়টি মূলত ইউজার ইনন্টারফেইস ডিজাইন সম্পর্কিত। ব্যবহারকারী আপনার সৃষ্ট এ্যাপলিকেশনের জানালাকে প্রয়োজনে বড়- ছোট করতে পারেন। জানালায় থাকতে পারে টেক্সট বাক্স, বোতাম ইত্যাদি। সুতরাং ফর্ম রিসাইজ হলে এগুলোও যাতে তার জায়গায় থেকে যায় সে ব্যবস্থা জরুরী। ডিজাইনকালে দু'টি প্রোপার্টি সেটিং দ্বারা তা হতে পারে। এগুলো হলো: ১. ডকিং ও অ্যাঙ্কোরিং। ডকিংয়ের কাজ, যে বস্তুর ওপর সে বসে আছে- যেমন ফর্ম, তার সাথে নিজের সাইজও বাড়ানো ও কমানো।

একটি কোডহীন উদাহরণ দ্বারা বিষয়টি বুঝিয়ে দিচ্ছি। ডানের চিত্রে আমরা একই ফর্ম রানকালে বড়ো ও ছোট সাইজে দেখতে পাচ্ছি। ফর্মের মধ্যে দু'টি অবজেক্ট আছে: বোতাম ও টেক্সট বাক্স। ব্যবহারকারী ইচ্ছে মতো ফর্মের সাইজ কমাতে বা বাড়াতে পারবেন। সাথে সাথে বোতামটি নিজের সাইজ অপরিবর্তিত রেখে স্বস্থানে থাকবে। অপরদিকে টেক্সট বাক্সের সাইজও প্রপোরশন্যালি বাড়বে কমবে। এই দু'টি ক্রিয়া নিশ্চিত করতে যেয়ে আমাদেরকে ডিজাইন টাইমে ফর্মের ও বোতামের Anchor নামক প্রোপার্টি সঠিকভাবে সেট করতে হবে।



টেক্সট বাক্সের উপর কোথাও ক্লিক করুন। প্রেপার্টিতে যেয়ে Anchor -এর ডানের এ্যারো চয়েজ করুন। ছোট্ট ৪টি চতুর্ভুজের রঙ গ্রে করুন একটা একটা করে ক্লিক করে [দেখুন নিচের প্রথম চিত্রটি]। এরপর বোতামে ক্লিক করে Anchor প্রোপার্টিতে যেয়ে এর শুধুমাত্র ডানের ও নিচের চতুর্ভুজ দু'টোর রঙ গ্রে করে দিন ক্লিক করে [দেখুন নিচের দ্বিতীয় চিত্রটি]। ব্যস! ফর্মের সাথে সঠিকভাবে টেক্সট বাক্স রিসাইজ হবে ও বোতাম তার জায়গায় থাকবে।

